

Web App **PENTESTING**

PenTest
magazine

Vol.2 No.3 Monthly ISSN 2084-1116
Issue 03/2012(05) March



Security Assessment of Web Services

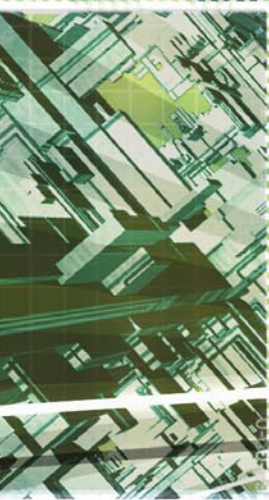
WEB APPLICATIONS WITH NO SECRETS
WEB SERVICES AND TESTING
BASIC DO-IT-YOURSELF WEBSITE SEO AUDIT
& OPTIMIZATION FOR SEARCH ENGINES
INTERVIEW WITH TOM BRENNAN
CHAPTER 4 OF CYBER STYLETTO

The Industry's First Commercial Pentesting Drop Box.

THE Pwn Plug.



Air Freshener?



Printer PSU?
...nope



FEATURES:

- ★ Covert tunneling
- ★ SSH access over 3G/GSM cell networks
- ★ NAC/802.1x bypass
- ★ and more!



PWNIE EXPRESS

@pwnieexpress.com

Discover the glory of
Universal Plug & Pwn

t) @pwnieexpress **e)** info@pwnieexpress.com **p)** 802.227.2PWN

Web Based CRM & Business Applications for small and medium sized businesses

Find out how Workbooks CRM can help you

- Increase Sales
- Generate more Leads
- Increase Conversion Rates
- Maximise your Marketing ROI
- Improve Customer Retention



AJ Thompson

Sales Director, Northdoor

We now have better visibility of business metrics, have streamlined our sales order processing and reduced our operational costs significantly.

Contact Us to Find Out More

+44(0)118 3030 100
info@workbooks.com

Dear Readers!

In this Pentest Magazine we prepared special combination of topics which, for sure, will interest you.

Let's take a closer look on what you can find there.

In the section Web services testing

If you go to page 6, you'll find there Rudra Peram, who is a Software Security Analyst and has over 10 years of experience in the field of Information Technology focusing on Web Application Security, Application Development and Software. In his article entitled: Security Assessment of Web Services he will guide us, among other things, through several ways of attacking web services.

In the next article, Jan will lead us, with examples, through popular web services with which we meet daily. For example social networks. Jan will finish his voyage on storage files in web cloud.

Right next to the Jan's article you will find something which gives you an overview in testing web services. In this article Malhotra will show you several forms of web services testing and will explain why and how we should test them.

The last but not least is the Basic Do-It-Yourself Website SEO Audit & Optimization for Search Engines, written by Monika Bańczerowska. At the end of this issue, we left for you something completely different – an article to teach a website's owner how to better score their website on the Internet.

Finally, at the end of this issue you will find an interview with Tom Brennan and fourth part of cyber crime novella-Cyber Styletto by Mike Brennan. If you are curious what will gonna happen – jump to the page 36 and enjoy reading.

We hope, you will find this issue of PenTest compelling and absorbing

Thank you all for your great support and invaluable help.

Enjoy the reading!
Paulina Plocha
& PenTest Team

WEB SERVICES TESTING

06 Security Assessment of web Services

By Rudra Peram

Web services which are designed primarily for systems to interact with each other and are not intended to be consumed directly by human beings. This assumption has severe consequences in several areas: developers are not as security conscious when developing web services. Negative testing of these services and Security teams are not focusing on these services either. The level of maturity of automated security testing tools for web services is not helping the situation either.

Important thing which author is touching are several ways of attacking a web services for example: authentication, authorization, SQL Injection, XPath and much more.

16 Web applications with no secrets

By Jan Pogocki

Times are changing, technology is changing too. Few years ago our computers were connected to the internet only across very slow dial-up modems. In this article-Web services with no secrets, Jan will help us to refresh beginning of technology and step by step will demonstrate how our lives are concentrated around broadband connection and access to many web-services- the world we cannot and don't want to imagine:)

22 Web services and testing

By Saurabh Malhotra

A web service is just a system which resides somewhere on a network and gives response specific requests from clients. Here you must be aware of the term clients, clients are not only the people working on their computers on the desk, and it can also be some machines.

BASIC

30 Basic Do-It-Yourself Website SEO Audit & Optimization for Search Engines

By Monika Bańcerowska

Building a completely search engine optimized website from scratch on an SEO friendly structure may seem an easier task than bringing an old one to work better with search engines. However, it is never too late to effectively optimize a website design for SEO to enjoy higher visibility and rankings, improve site's credibility, get pages indexed smoothly and stop missing out on traffic.

INTERVIEW

32 Interview with Tom Brennan

By Aby Rao

Tom took a front row seat on the architecture, development, administration and security of computer-controlled systems with experiences ranging from the financial trading floor of Wall Street to the United States Marines Corps.

CYBER STYLETTO

36 Cyber Styletto

By Mike Brennan and Richard Stiennon

Cyber crime novella- Cyber Styletto – Chapter 4



TEAM

Editor: Paulina Plocha
paulina.plocha@software.com.pl

Betatesters: Denis Distler, Felipe Martins, Rishi Narang, Johan Snyman, Edison Josue Diaz, Aby Rao, Hugo Lujan

Senior Consultant/Publisher: Paweł Marciniak

CEO: Ewa Dudzic
ewa.dudzic@software.com.pl


Art Director: Ireneusz Pogroszewski
ireneusz.pogroszewski@software.com.pl
DTP: Ireneusz Pogroszewski

Production Director: Andrzej Kuca
andrzej.kuca@software.com.pl

Marketing Director: Ewa Dudzic
ewa.dudzic@software.com.pl

Publisher: Software Press Sp. z o.o. SK
02-682 Warszawa, ul. Bokszerska 1
Phone: 1 917 338 3631
www.pentestmag.com

Whilst every effort has been made to ensure the high quality of the magazine, the editors make no warranty, express or implied, concerning the results of content usage.
All trade marks presented in the magazine were used only for informative purposes.

All rights to trade marks presented in the magazine are reserved by the companies which own them.
To create graphs and diagrams we used [smartdraw.com](http://www.smartdraw.com) program by  SmartDraw

Mathematical formulas created by Design Science MathType™

DISCLAIMER!

The techniques described in our articles may only be used in private, local networks. The editors hold no responsibility for misuse of the presented techniques or consequent data loss.

Security Assessment of Web Services

Web Services are designed primarily for systems to interact with each other and are not intended to be consumed directly by human beings. This assumption has severe consequences in several areas: developers are not as security conscious when developing web services, QA's are not as focused on performing Negative testing of these services and Security teams are not focusing on these services either.

The level of maturity of automated security testing tools for web services is not helping the situation either.

All major organizations have their presence on the web and Web Services power all of the AJAX and other Web 2.0 applications. In several instances, these web services were never intended to be accessible from the internet. Existing services that serve data to internal applications are reused to serve applications on the internet. Coupled with the fact that web services don't receive the level of scrutiny (as web applications) from security perspective, a lot of services are vulnerable and ripe for exploitation.

In this article, we will look at the various vulnerabilities that can manifest in web services and go over some test cases that can help us to identify these vulnerabilities.

Web Services are invoked by web applications over either HTTP/HTTPS protocol similar to web application and this makes it easy for anyone to call these services directly without having to go through the intended web applications. Web services exhibit same vulnerabilities that are similar to web applications and some vulnerabilities are unique to the services realm.

SOAP based Services

SOAP based services utilize a combination of XML, WSDL, SOAP & UDDI technologies to provide the necessary services.

- *Web Services Description Language (WSDL)* is used to describe all the available services and their individual signatures.
- *Universal Description, Discovery and Integration (UDDI)* acts as a registry to register individual web services and allows potential clients to discover these services. Several organizations provide the WSDL URL to identify all available services instead of utilizing UDDI Registry.
- *Simple Object Access Protocol (SOAP)* technology provides the mechanism to package the payload between web services and their clients
- *Extensible Markup Language (XML)* is used to represent the request and response between the services and the clients.

For demonstrating the various vulnerabilities and ways to identify them, I will be using OWASP WebGoat [1] application. This application provides a great learning opportunity to identify and fix all the common (and not so common) vulnerabilities in web applications and web services.

Identification of Service Endpoints

In testing web services, the first step is to access the WSDL file to identify all the available services and the types of inputs/outputs accepted by these services.

Given this WSDL endpoint for OWASP WebGoat web service: `http://localhost/webgoat/services/SoapRequest?WSDL`, you will get the following response (Listing 1).

This WSDL indicates the presence of `getCreditCard` method which takes an `id` as an Input. Armed with this information, this service is ready to be tested. Test can

be conducted against this service in a number of ways. For those who prefer command line option, CURL [2] can be used to test this service.

```
curl --request POST --header „Content-type: text/xml“  
--data @my_request.xml http://localhost/webgoat/  
services/SoapRequest
```

Listing 1. Contents of a WebGoat WSDL file

```
...  
<wsdl:portType name="WSDLScanning">  
  <wsdl:operation name="getFirstName" parameterOrder="id">  
    <wsdl:input message="impl:getFirstNameRequest" name="getFirstNameRequest"/>  
    <wsdl:output message="impl:getFirstNameResponse" name="getFirstNameResponse"/>  
  </wsdl:operation>  
  <wsdl:operation name="getLastName" parameterOrder="id">  
    <wsdl:input message="impl:getLastNameRequest" name="getLastNameRequest"/>  
    <wsdl:output message="impl:getLastNameResponse" name="getLastNameResponse"/>  
  </wsdl:operation>  
  <wsdl:operation name="getLoginCount" parameterOrder="id">  
    <wsdl:input message="impl:getLoginCountRequest" name="getLoginCountRequest"/>  
    <wsdl:output message="impl:getLoginCountResponse" name="getLoginCountResponse"/>  
  </wsdl:operation>  
  <wsdl:operation name="getCreditCard" parameterOrder="id">  
    <wsdl:input message="impl:getCreditCardRequest" name="getCreditCardRequest"/>  
    <wsdl:output message="impl:getCreditCardResponse" name="getCreditCardResponse"/>  
  </wsdl:operation>  
</wsdl:portType>  
...
```

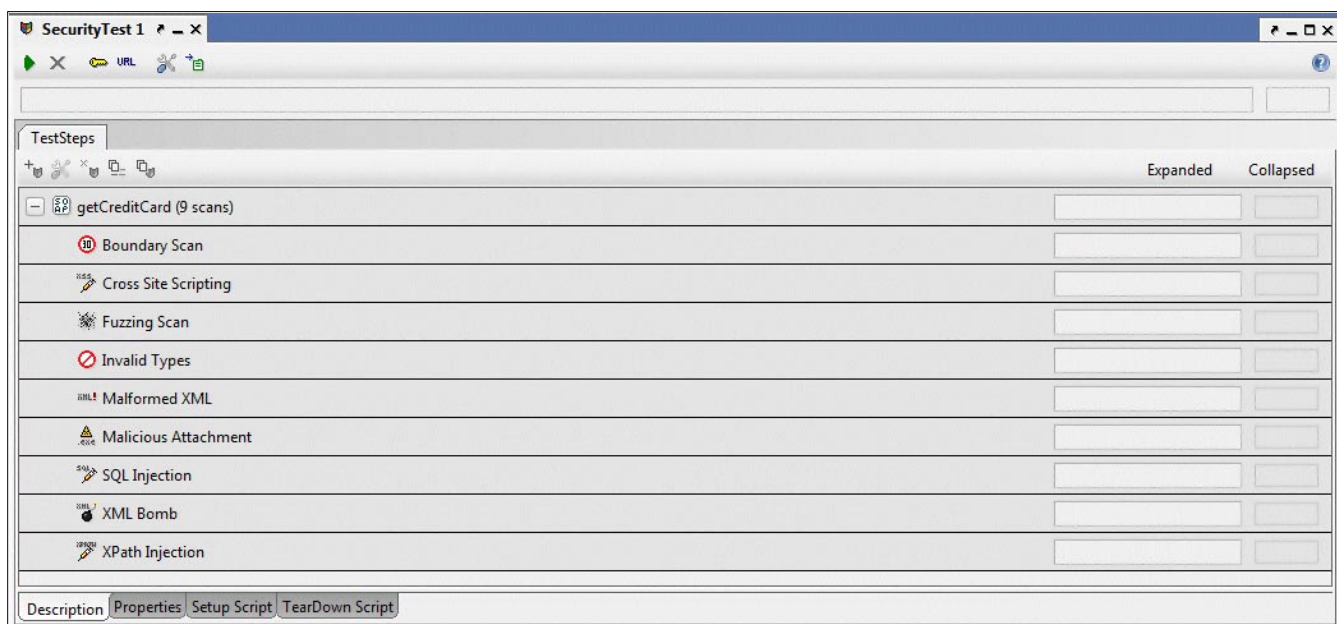


Figure 1. SoapUI screen showing the list of all available security tests

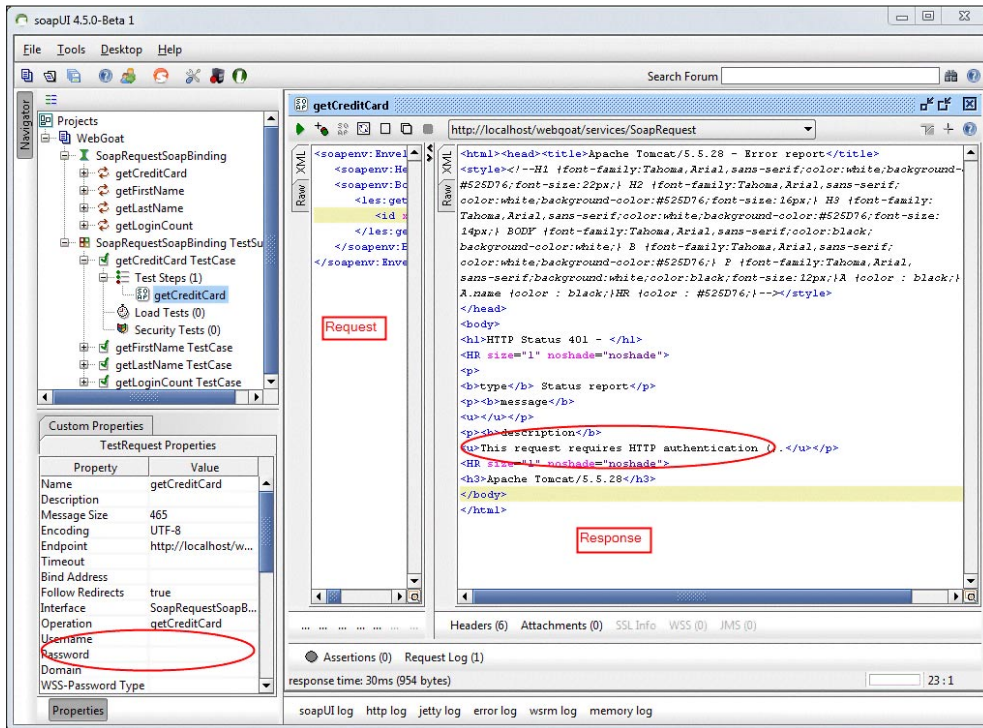


Figure 2. Accessing a web service without providing Authentication Credentials

There are several tools to conduct web service testing. SoapUI, WSDigger, WSFuzzer, HP WebInpsect, IBM AppScan are some of the well-known tools used for Web Service security testing. For this article, I will be using the open source version of SoapUI [3].

directly once they gather all the necessary information for invoking these services from the WSDL file.

Web services must enforce that users are authenticated before serving their requests without assuming that requests can only originate from web application after authenticating users.

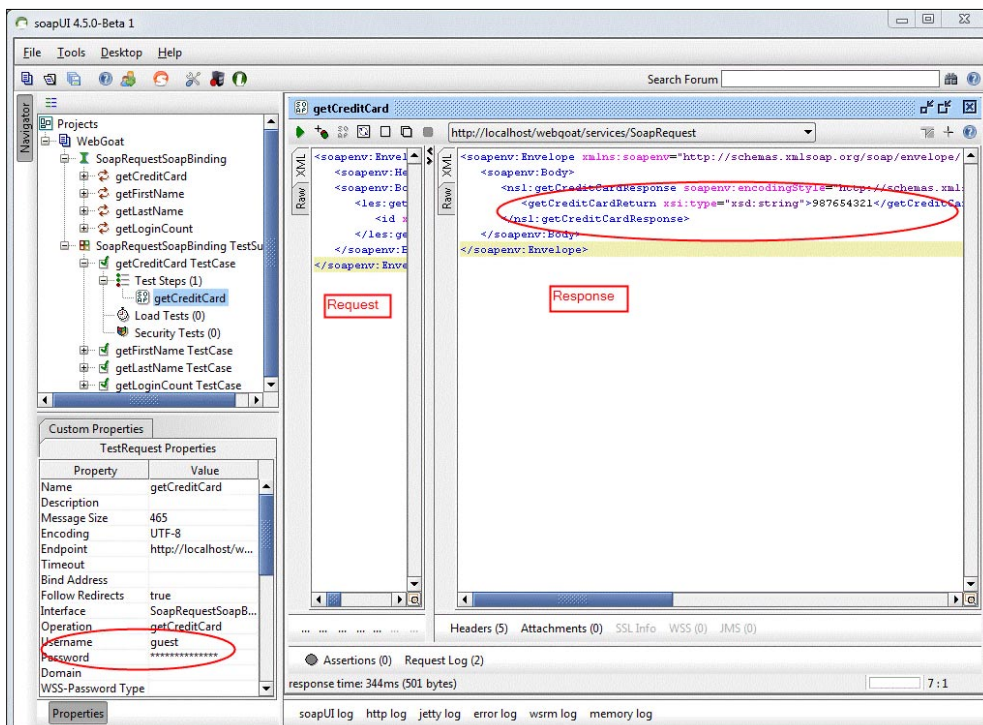


Figure 3. Accessing the same web service after providing Authentication Credentials

Attack Scenarios

There are several ways of attacking a web service. Let's go over some of the test cases here. As an example, here is a snapshot showing the different types of attacks that can be performed on web services using SoapUI.

Attack Scenario 1:

Authentication

Developers might make assumptions about how web services will be invoked without enforcing rules for invoking the services. In normal scenario, web services are invoked by web applications when serving functionality for authenticated users. However, nothing prevents an attacker from invoking these services

Figure 2 shows that the Apache Tomcat web server is blocking the request in the absence of HTTP Basic authentication credentials. However, when we pass the right credentials, the web service is sending us back the appropriate response as shown in Figure 3. The difference in response between these two requests indicate that authentication is enforced for the `getCreditCard` web service.

Attack Scenario 2:

Authorization

Once a user is authenticated by the system, Authorization checks are performed as and when users invoke any services to ensure

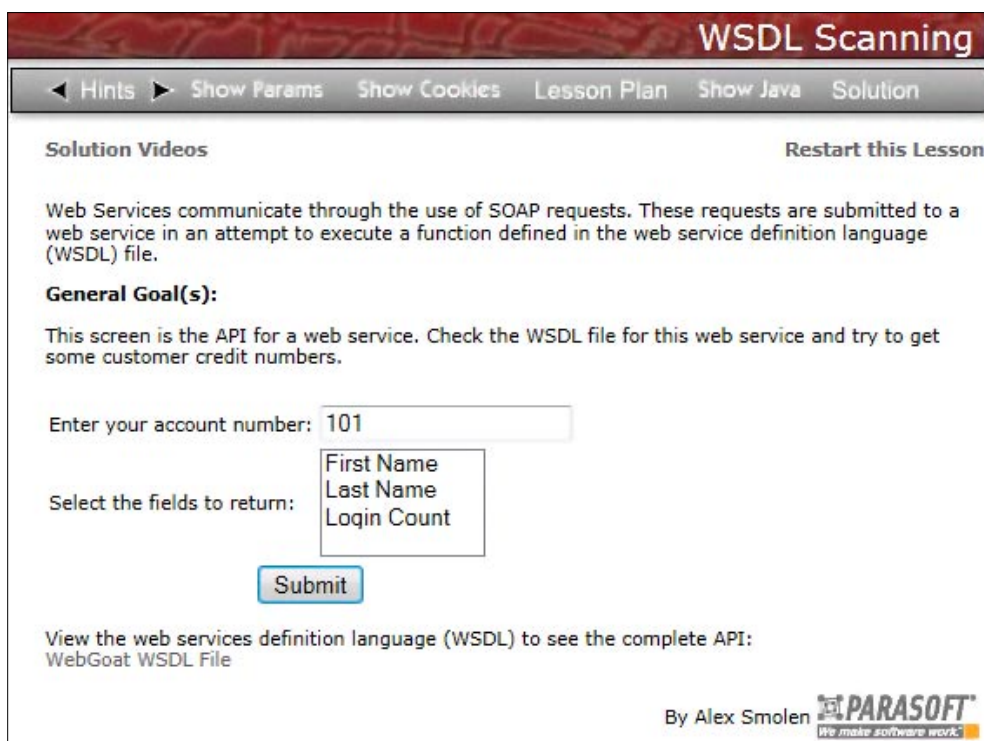


Figure 4. WebGoat UI listing options that invoke only 3 operations listed in the WSDL file

that access to these resources are provided only to authorized users.

We have already seen that WSDL files present a list of all web service methods available for consumption. Different web applications developed in the same organization might invoke different subsets of the web services published in a common WSDL file.

Listing 1 indicates that there are 4 different web services listed in WebGoat WSDL file however Figure

4 shows that the WebGoat application provides only 3 of these services from within the application. This potentially indicates that either the user that is logged in to WebGoat doesn't have the necessary authorization to access the `getCreditCard` service or that this service is not intended for that application.

Using a proxy or a web service testing tool like SoapUI, we can craft payloads to invoke the services that are not available from within the application. In this scenario, we use SoapUI to create a request to invoke `getCreditCard` service.

We were able to access the Credit card number stored in this User profile by making a direct call to the web service even though the web application does not provide this functionality. If the logged in user was not supposed to invoke the `getCreditCard` service as indicated by the WebGoat application, then we have identified a failure in enforcing proper authorization checks before allowing the user to invoke the web service.

Attack Scenario 3: SQL Injection

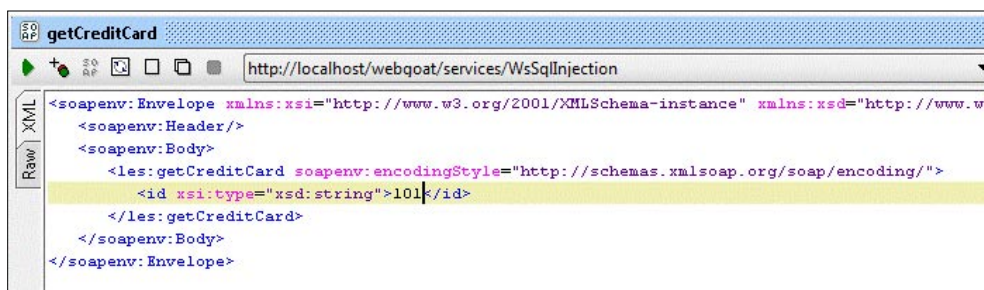


Figure 5. Using SoapUI to create a request for `getCreditCard` service for `userId=101`

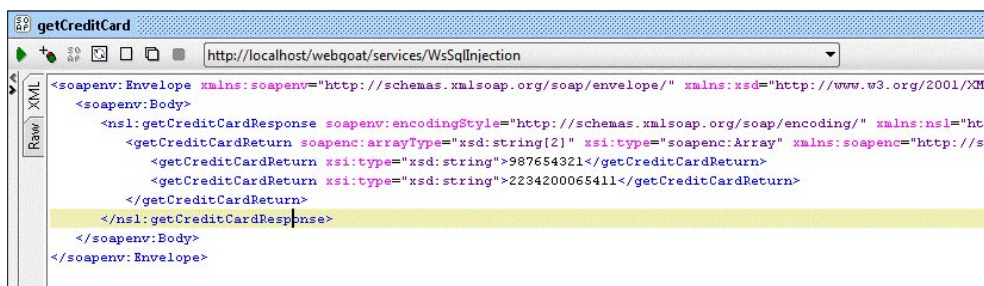


Figure 6. SOAP Response returning the Credit card values for `userId=101`

SQL Injection is one of the classic examples of the vulnerabilities that can occur when user inputs are used within an application without proper validation. SQL Injection can allow an attacker to gain complete control of the database and all the information stored in it.

Here is a common way of verifying user credentials from the database in an insecure manner.

```
"Select * from users where
UserId = '' + userId + ''
and Password = '' + pwd +
'';"
```

This code take user input for `userId` and `pwd` fields and concatenates them to a static query without performing any validation on the user input.

In this test case, we will send multiple requests to identify whether the web service is vulnerable to SQL Injection or not. If we craft a request payload as shown Listing 2.

This request will result in a query as shown below:

```
"Select * from users where UserId = '' and Password = '';"
```

Listing 2. Attack strings to verify the existence of SQL Injection vulnerability

```
<login>
  <userId>'</userId>
  <pwd>'</pwd>
</login>
```

Listing 3. Attack strings that will bypass Authentication in case SQL Injection vulnerability

```
<login>
  <userId>' or 1=1--</userId>
  <pwd>' or 1=1--</pwd>
</login>
```

Listing 4. XML document storing all User information

```
<users>
  <user>
    <name>John Doe</name>
    <email>john.doe@example.com</email>
    <userId>jdoe</userId>
    <password>test1</password>
    <role>admin</role>
  </user>
  <user>
    <name>Bill Gates</name>
    <email>bill.gates@example.com</email>
    <userId>bgates</userId>
    <password>test2</password>
  </user>
  <user>
    <name>Steve Jobs</name>
    <email>steve.jobs@example.com</email>
    <userId>sjobs</userId>
    <password>test3</password>
    <role>user</role>
  </user>
</users>
```

If the system is vulnerable to SQL Injection, we will get a detailed error message about the database and the error code and other useful information that will help us fine tune our attacks.

In our next step, we will enter the value of `'` or `1=1--` for both the username and password elements.

This request will result in a query as shown below:

```
"Select * from users where UserId = '' or 1=1--
and Password = '' or 1=1--"
```

When this query gets executed, it will return all the rows in the users table and in most cases the user will be logged in as the first user in the database which will be an administrator account.

Attack Scenario 4: XPATH Injection

XPath Injection is very similar to SQL Injection in that unvalidated user input is used to construct a query to search data in XML documents. XPath can be considered as the SQL equivalent for XML.

Let's assume that the system contains a XML document of all the users and their credentials as shown Listing 4.

Here is a typical XPath query to authenticate users based on the structure of the XML datastore shown above.

```
String("/user[userId/text()='\" + request.get("userId") +
  \"' and password/text()='\" + request.get("password") + \"']")
```

Similar to SQL Injection, we will send a few different requests to identify whether the service is vulnerable to XPath injection.

First, we will send invalid node IDs to check whether we can get more information about the XML parser that is being used. In the example shown below, we are passing `uid` and `pwd` instead of `userId` and `password`.

```
String("//user[uid/text() = 'junk' and pwd/text() =
  'morejunk' ])
```

If the web service doesn't present custom errors with generic message, we will get back detailed error message when the service encounters invalid nodes.

In our next test case, we will send a request to bypass Authentication. We use the characters `"--"` to comment out the rest of the query in SQL Injection. However, there is no equivalent for `"--"` characters in XPath Injection. So, we need to modify the attack string such that the query will evaluate to true even in cases where we don't have a valid username and password. To achieve this requirement, we will send the following

attack string (`'junk'` or `1=1` or `'a'='b'`) for the `userId` value resulting in the following XPath query.

```
String(//user[userId/text() = 'junk' or 1=1 or 'a'='b'
and password/text() = 'morejunk'])
```

Due to higher precedence of AND over OR in logical evaluations, the parser will evaluate the (`'a'='b'` and `password='morejunk'`) part first resulting in false and then the parser will evaluate the (`'userId=junk'` or `1=1`) part to true and the OR operation of those two evaluations will always result in true and the attacker will be authenticated.

In our next test scenario, an attacker can try to guess the structure of the XML document by changing the `1=1` part in the attack string. With the attack string shown

below, the attacker is trying to verify whether the first sub-node is named as `name`.

```
'junk' or name(//users/name[1]) = 'name' or 'a'='b'
```

If the first node is called as `name`, then the attacker would be successfully authenticated. If the first node is not called as `'name'`, then the user will not be authenticated indicating the attacker to guess for some other node name. By automated this process of guessing, the attacker might be able to guess the entire structure of an XML document.

Attack Scenario 5: Cross Site Scripting (XSS)

Cross site scripting is the most common type of vulnerability present in all web sites today. This attack

Listing 5. Order of Evaluation of the XPath by the XML parser

```
String(//user[
userId/text() = 'junk' or 1=1          - Evaluated Second
or
- Final Evaluate Step of ORing the results of first two evaluations
'a'='b' and password/text() = 'morejunk' - Evaluated First due to the presence of AND operator
])
```

Listing 6. Sending a Script instead of UserId to test XSS vulnerability

```
<login>
  <userId><script>alert("test")</script></userId>
  <pwd>test</pwd>
</login>
```

Listing 7. Extra Script node within userId nodes

```
<login>
  <userId><script>test</userId>
  <pwd>test</pwd>
</login>
```

Listing 8. Missing userId end node

```
<login>
  <userId>test<userId>
  <pwd>test</pwd>
</login>
```

Listing 9. Missing userId end node

```
<login>
  <userId>test
```

```
<pwd>test</pwd>
</login>
```

Listing 10. Missing userId start node and addition of a random script node

```
<login>
  <script>test</userId>
```

```
</login>
```

Listing 11. Missing userId start node

```
<login>
  test<userId>
  <pwd>test</pwd>
</login>
```

Listing 12. Incorrect nesting of pwd and login nodes

```
<login>
  <userId>test<userId>
  <pwd>test</login>
</pwd>
```

injects a script in to website and targets the users of that particular website. Similar to SQL & XPath injections, the root cause of this issue is unvalidated User Input being processed by the website.

Some people consider web services to be immune from XSS issues since the data is returned as XML instead of HTML. However, every component of a web application stack (including the front-end code, server-side code and web services) must validate user input data before storing it in the database or including the data in the response that is being sent back to the user.

Here is a simple test case to determine if the system is susceptible to XSS attacks.

In this test, the attacker is sending a simple script in the username field for authentication. If the attacker gets an error message saying the given user does not exist and the actual script gets returned as is in the user name information, this indicates that the web service is vulnerable to XSS attacks.

To learn more about the different types of XSS attacks, please visit OWASP website [3]. Also, a comprehensive library of XSS attack strings is available at RSnake's website [4].

Listing 13. Repeating userId node with large value for each userId node

```
<login>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  <userId>I am an a very large String (10 MB)</userId>
  .....
  <pwd>test</pwd>
</login>
```

Listing 14. Dereferencing 'hi30' will perform DOS attack on the system

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hello [
  <!ENTITY hi0 "Hello World">
  <!ENTITY hi1 "&hi0; &hi0;">
  <!ENTITY hi2 "&hi1; &hi1;">
  <!ENTITY hi3 "&hi2; &hi2;">
  <!ENTITY hi4 "&hi3; &hi3;">
  ...
  <!ENTITY hi30 "&hi29; &hi29;">
]>
<hello>&hi30;</hello>
```

Listing 15. EICAR test virus being uploaded to FileUpload service

```
<FileUpload>
  <filename>eicar.pdf</filename>

  <data>X50!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*</data>
</FileUpload>
```

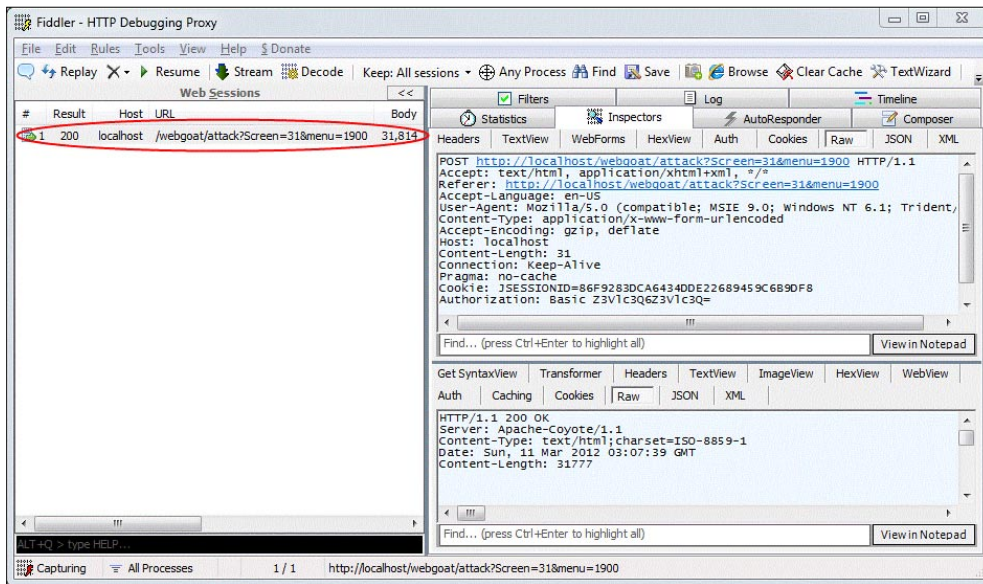


Figure 7. Using a Proxy (Fiddler) to capture the request to the web service

Attack Scenario 6: Malformed XML

In this test case, the goal is to send malformed XML and elicit useful error messages or to cause unexpected behavior. XML documents must be well-formed to be processed successfully unlike HTML documents and this behavior can be used to gather more information about the system. It is very easy to generate tons of malformed XML tests. A few of them are shown Listing 7-12.

I will leave the task of generating more malicious attacks up to imagination of the person conducting the web service assessment.

Attack Scenario 7: Denial of Service

Denial of service affects the availability of the service for real business users and can adversely affect the

business operations resulting in loss of revenue and reputation of the company. This attack focuses primarily on preventing the availability of the service to real users and usually does not affect the confidentiality or integrity of the data.

In this attack, we exploit the same requirement of XML documents needing to be well-formed to function properly. When a XML payload reaches the server, the parser has to read through the entire XML document to identify whether the XML document is well-

formed or not. XML document can be created in such a way that the parser will consume all the CPU & memory resources at the server effecting creating a Denial of Service and eventually bringing down the server.

DOM based parsers are inherently vulnerable to DOS if an attacker send a huge XML document due to the need for these parsers to load the entire XML document in memory before parsing it. SAX based parsers don't have this vulnerability as they read the data from file and parse it in parallel without having to load everything in memory.

In the example shown below, the userId node contains a large amount of data and the same node is added multiple times to create an extremely large XML document.

When a parser evaluates this document, it ties up a lot of server resources and we can bring down the server by sending a few of the same requests until the server becomes non-responsive.

Another well-known attack for DOS is known as XML Bomb. XML Bombs are specially crafted XML documents that exploit the fact that data entities can be defined as part of the XML document.

If the "hello" node gets dereferenced, it would take a lot of CPU resources to compute the value of hi30

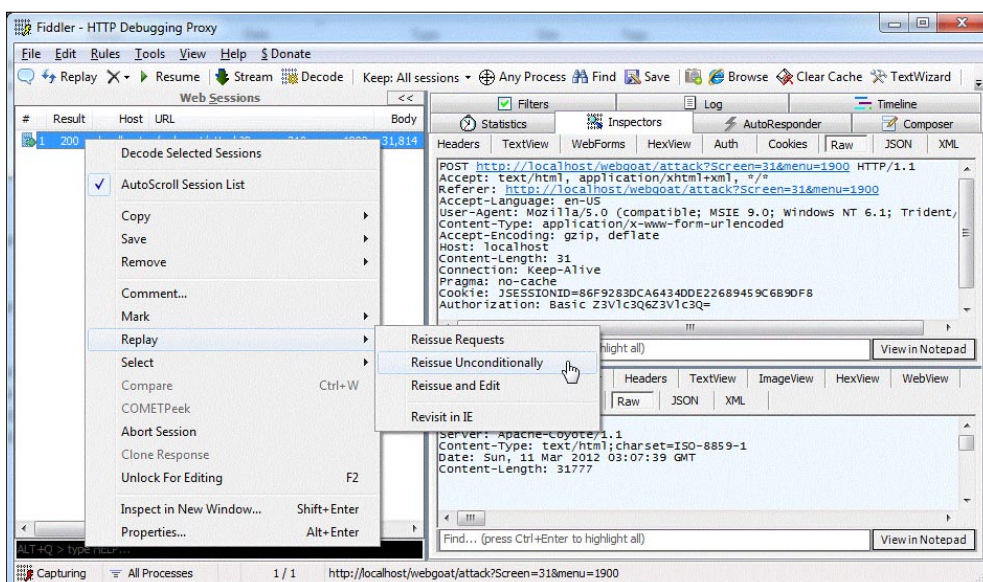


Figure 8. Replaying the captured Request

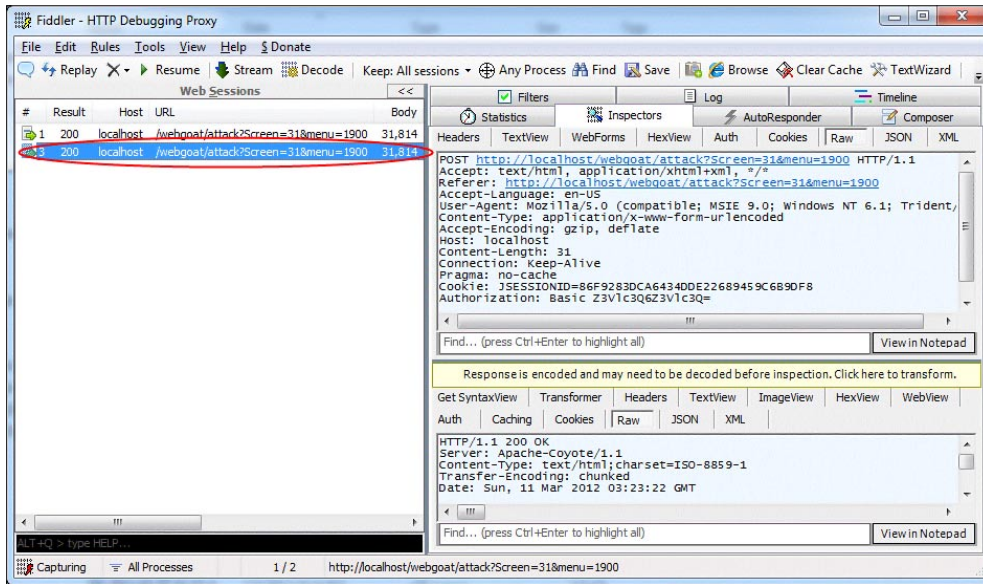


Figure 9. Server sent the same response for the replayed request (Response size = 31814 bytes for the original and replayed requests)

and would block the entire server from any other processing. When the value of hi30 gets computed, it would take 11GB of RAM to store the entire value thus essentially either crashing the server or making it unresponsive.

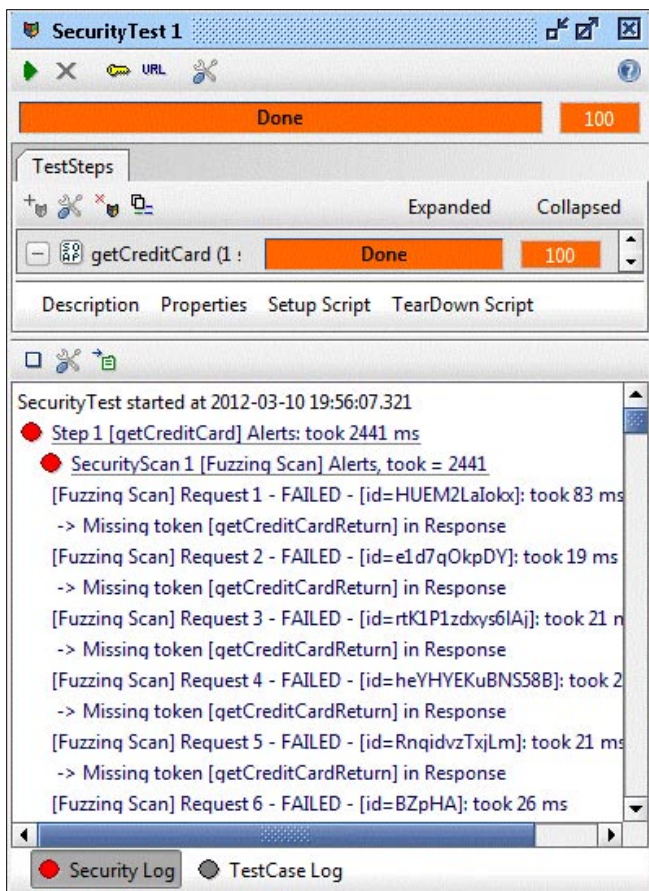


Figure 10. Using SoapUI for fuzzing the GetCreditCard Service

Attack Scenario 8: Malicious Attachments

When a web service accepts an attachment as part of its payload, tests must be conducted to ensure that the attachments are scanned to identify any virus or malware.

Upload the EICAR virus [6] to your web service and verify that the file is scanned for virus and malware before the file gets processed by the web service.

The content of the *data* node indicates that it is a test virus and this signature is added to the virus signature set in all common anti-virus

systems. So, all these anti-virus systems will flag this file as infected with virus if the file contains only the data specified in the chunk node.

Attack Scenario 9: Replay attacks

In a replay attack, an attacker sniffs the traffic between the client and the server and replays the traffic to the server at a later time. In the absence of timestamps and other sequences numbering controls, the server has no way to determine if the request is actually coming from the client or is being replayed by an attacker. Replay attacks allow the attacker to assume the identity of a user without knowing their credentials.

An attacker would use a Network sniffer like Wireshark to capture packets over the wire and use a tool like TCPReplay to replay the traffic or manually create a request with the same data using a proxy. For simplicity purposes, we will be using Fiddler proxy [7] to capture the request and replay the same request. If the server provides the same response to the original request and the replayed request, then the system is susceptible to replay attacks.

Attack Scenario 10: Fuzzing

Fuzzing is a process of using an automated tool to send multiple requests to the same service with minor changes in the data. This dataset is designed in such a way that it includes data that has the potential to trigger anomalies in the processing application. Most of the fuzzing inputs would involve providing invalid, unexpected or random data.

If a program expects an input integer, a fuzzer would send string, boolean or other invalid data types; singled

References

- OWASPWebGoat–https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project [1]
- CURL – <http://curl.haxx.se/> [2]
- SoapUI – <http://www.soapui.org/> [3]
- OWASP – <http://www.owasp.org/> [4]
- XSS Cheat sheet – <http://ha.ckers.org/xss.html> [5]
- EICAR Test Virus – <http://www.eicar.org/86-0-Intended-use.html> [6]
- Fiddler – <http://fiddler2.com/fiddler2/> [7]

int, unsigned int, float, double etc., for unexpected data and random gibberish that includes data from different datasets. When a program receives these types of inputs, there is a tendency for the program to crash or hang or get in to some sort of data overflow situation and would allow an attacker to exploit the system.

There are several open source and commercial fuzzers available in the market to automate the fuzzing process. Some of the well-known fuzzers include Sulley, JBroFuzz, WSFuzz, Peach, etc.,

Figure 10 shows that SoapUI fuzzing module is sending a bunch of gibberish for Id element and indicates that all those tests are failing normally without hanging or crashing the server.

Conclusion

Web Services are being consumed more widely from client browsers with the advent of AJAX and other Web 2.0 technologies. In this article, we have seen several ways in which these web services can be assessed to verify the security of these services. Wide spread usage of these testing activities will improve the security of these services and ultimately the security of the web applications these services were designed to serve. ♦

RUDRA PERAM

Rudra Peram is a Software Security Analyst at Apollo Group. He has over 10 years of experience in the field of Information Technology focusing on Web Application Security, Application Development and Software Quality Engineering.

Join

PenTest Mag team!



PenTest Magazine is looking for regular contributors. If you want to be a part of the first magazine devoted to penetration testing, now's your chance to join us. We especially need:

- news contributors – send in a piece of news of an interest for a pentester and make your own comment on it.
- “point of view” section writers – short articles (800 words tops) with you discussing an issue you think should be discussed.
- “vulnerability check” writers – what a pentester can use in his work.
- reviewers – found an interesting tool? Review it for us.
- betatesters – read an article before it's published in the magazine and share your opinion on it with us.

Regular contributors are given free subscription to the magazine and – if they represent companies – free advertising in the mag. And, of course, an earned mention in the magazine.

Worth it? Ask for details:

maciej.kozuszek@software.com.pl

Web Applications with no Secrets!

Times are changing, technology are changing too. Few years ago our computers were connected to the internet only across very slow dial-up modems. Now, we can't imagine world without speed broadband connection and access to many web-services.

If few years ago somebody would have asked me, which web-services I was using, I would answer that only electronic mail and some news portals. I remember times when simple e-mail web clients were appearing. It was something new, because everyone who had was used to using desktop mail clients, like an Outlook Express. Some difficulties were appeared when you want to check up your mails at other computer, without your Outlook's default configuration. Smartphones, tablets or different mobile devices were not in common as now; laptops were very expensive and unprofitable to buy. I did not mention yet about dial-up internet connection, with characteristic sounds coming from speaker localized inside the computer. Moreover it was very slow and luxurious connection, not available everywhere. Dial-up internet was a big barrier to develop it.

Internet evolution immediately started together with broadband connection. Providers offered twice or more better speed. It was impulse to action. Very quickly old Internet Explorer browser was replaced (of course not at all) by other competitive browsers, especially by brand new Mozilla Firefox, which main idea were based on compatibility with new internet standards. We could not tell that about Internet Explorer, because many years later Microsoft still insisted at their "standards". Webmasters had big quandary: write websites compatible with Internet Explorer or with W3C

standards and other browsers? But returning to main topic: over time websites had become more and more interactive and attractive for users. That new creature had been called web applications. It was entailing with appearing new languages and techniques creating websites. I think neither Flash nor Java applets were such innovative like AJAX. Nowadays, on the one hand it is used by the largest companies in their products and on the other hand by simple bloggers in their *e-diaries*.

I am laughing that now people must have only web browser with connection to the internet even without operating system. Of course there is for example Google Chrome OS – web browser as an operating system, but now it is not yet time to off-load our lovely Windows. Nevertheless web services dominated our lives spending in the front of computer or other mobile device with internet access.

Main web application what we are in the majority using is webmail. All of today's primary mail service providers usually give us access to our account in three ways: POP3, IMAP and exactly webmail. Many people think the last one way is the best and they are using it. Why? The first advantage is ease of using webmail. Wherever on the world you open any browser you will see the same content and your personal settings. It is easier than every time configuring mail client. Anyways you may only want

to check your mail in short time, which not allow to do all this configuring process. The second advantage is expressed in safety of using. For example Gmail from Google optionally offers two-step verification. It means that knowledge the password is not enough for potential hacker. The thing is that to get access our mails extremely important is specially generated code, which is sent by text message to our mobile phone. I think it is the greatest way to guard private correspondence against intruders. Next advantage is possibility of syncing our settings and contacts on all our computers and mobile devices with special e-mail applications. Keeping this data in mind is very difficult; on the other hand writing that in something like notebook is outdated. Syncing helps us out. Moreover majority of modern web mail clients offer us a chat, like a Google Talk based on Jabber. Chat client in browser allows bring it with all conversation archive wherever we want by the laptop, netbook or other mobile device.

Next very important web service we are used to use is online banking. Now, we cannot imagine our business and personal payments without possibility to

do that on the internet. E-banking changed our lives. Quick money transfers, opening deposit accounts, finally checking amount of balance when only you want and from where you want – there are main undisputed advantages, which are enforcing to open bank account with access to the internet banking. Now you do not have to queue and wait for your bank assistant to do money transfer. Using bank web application is very simple. Many even elderly people are managing their money this way, so I think it denote correctness of my ascertainment. Today, banks offer very advantageous conditions to maintenance web accounts. Remain only aspect of safety our money. Of course hacking e-account and theft money still exist, but above all, risk all the time is going down. Largely it is our fault. Real bank never send us messages with request to enter on their website our login and passwords. If we did that, we should immediately call the bank to block our account or... prepare ourselves to unexpected withdraws. But generally banks encrypt connection in web browsers with strong certificates and in case doing something in account we have to confirm that entering password read from token, tangible list of disposable codes or

a d v e r t i s e m e n t



Web Based CRM & Business Applications for small and medium sized businesses

Find out how Workbooks CRM can help you

- Increase Sales
- Generate more Leads
- Increase Conversion Rates
- Maximise your Marketing ROI
- Improve Customer Retention

Contact Us to Find Out More

+44(0) 118 3030 100

info@workbooks.com



from mobile text message. It makes a lot of difficulties to take over our money by hackers.

Other very interesting example of very popular web services is social networks, like Facebook, Twitter or Google+. Over last five years this kind of web applications strongly gained ground. Many people got caught this huge eddy and now they are addicted of social networks. It gives us a lot of messages from our friends, what are they doing now. Human is very nosy being, he likes to know all about his nearest (but not only!) society. This web services hit the spot. Twitter is a microblog. The thing is, you write your thoughts only in 140 characters. Your message can be read by every single person all over the world, but in particular you associate *followers* around you. Each of them will see your *tweet* on their virtual dashboards. Each of them can reply on your message, start conversation with you or *retweet* your 140-characters information, what means – share it to their *followers*. This way you can get more people who want to read your posts. Any benefits? Yes, of course. If you have a big amount of people, who are observing you, you can invite them to your website or web application. Twitter web interface is very simple, but it has a lot of Java Script, AJAX implementations, as well as Facebook and Google+. But it is different story. History of Facebook begins in 2004 when Mark Zuckerberg, now, he the richest young man in the world, started project named The Facebook. The way of working it is very simple: people are publishing posts what and where are they doing now; optionally they can add one or more photos or videos which show this situation. Facebook is a mirror reflection of our real life: there are our friends, relationships, text chats, audio or video conferences based on Skype's engine, fan or anti-fan pages of corporations, trademarks, known people (politics, journalists, celebrities and people from the first pages of newspapers). Facebook is a huge database about over 845 million people! Soon if amount of active users grow up, Facebook will store all information about 1/7 human race society: from name, surname, date of birth and finishing with politics view, religious view, private messages to friends and posts on the Facebook's tables. Facebook is making conditional like a narcotic. After few days using this social network web service, every day, first in the morning, using computer, laptop or other device, we have to check up main stream, what's up in my friends? Did somebody change his relationship status? Or maybe appeared new photos from our yesterday's party? Facebook is a second life and that is engrossing. Recently I read the article that said people's productivity decline when they are browsing social networks. I even came

across with people, who cannot imagine world without access to Facebook. At home, all the time they are staring on the monitor and following each friend's move. On the free air, at school, on the shop – in a word: everywhere except behind computer – they are using Facebook application for smartphones and other mobile devices. Few months ago Facebook introduced new functionality named *Timeline*. *Timeline* is an axle of our life from birthday to nowadays. It shows our life in good light, how we had born, how was our childhood, what were we use to do etc. In the internet we can make ourselves a star, on Facebook too. It applies *Timeline* of course. Therefore when it was introduced, psychologist were reporting, that there is lots of instances people who was extremely jealous of friends, who had better biography, better told and illustrated. One more danger comes from using Facebook and generally internet – loneliness. At first sight it is a paradox. Loneliness, when we have over half thousand friends? Yes, it is truth, because then we are not focusing on real relationships, we are talking to friend less, successively we are separating from real world, what leads to personality disaster. Turning back to technology aspect of Facebook I can say this big society platform is wonderful, because each of us who have account there, can use it as a global ID on many, many websites. Now, we do not have to enter our nickname, e-mail address and password all the time, verify this data by clicking some link on the confirmation e-mail when we want to register on some website. We have to only click *Connect with Facebook*. We do not have to remember all this passwords, because logging to website is limited to click the same Facebook button. That is great connection Facebook – the biggest database about us – and Open ID – prototype of *Connect with...* button. Otherwise Facebook allowed bringing closer producers and consumers, celebrities and fans. Formerly contact with that two groups were very difficult.

Next very interesting web service, which started in the beginning of 2005, is popular YouTube. YouTube is a place, where everybody can share his video. It could be birthday party, interesting nature phenomenon, how-to video – everything. In this way we can simply share video with our family, friends or all the YouTube community. First idea of this web service was to share short, up to 10-minutes videos on low resolution. Over time YouTube came through a big siege – a lot of people wanted to upload their video and share it with each other. It led to improve service's capacity by developing new infrastructure. Interest of YouTube was such huge that finally, in 2006 Google decided to take it over. Immediately service was developing and

improving. Next, YouTube changed default videos aspect ratio from outdated 4:3 to panoramic 16:9, added HD 720p, Full HD 1080p and even 3D videos supporting. Time limit was extended firstly from 10 minutes to 15, now there are no time limits. Size of video is not important too, because now YouTube accepts each video without regard for its size. YouTube formed a new kind of blogging by recording quick movies. It is called vlogging. Vlogging is very popular way to give emotions and sense of speech. To be a vlogger we have to have only webcam with microphone and a lot of enthusiasm. Using YouTube is very simple; we can create new playlists and add them videos which we want to see. YouTube shared applications for mobile devices, so we can watch all video from YT Database for example on the underground, in a train, at school. Moreover most of new TV manufactures include YouTube application, so we can watch videos as well on our TV sets. YouTube allows sending videos by website or mobile devices. Second solution is very useful when we have not access to computer. Hosting that huge project request special attention, because lots of sent videos violate the law and copyrights. YouTube worked out mechanism which immediately recognize protected soundtracks in the movie and flag it. In majority cases YouTube marks film as controversial, because it is using third-party content and adds advertisements in the player. Otherwise YouTube can mute our film or completely block access to it and close our account, named channel. Now, YouTube has got 800 million unique users every month, which sends 48 hours of films every minute, what gives 8 years films every single day. It is amazing figures. This web service definitely will be remembered in a future as first, the biggest place where everyone could upload and share his own film with no time and size limits.

Other interesting web services are GPS sport tracers like Endomondo. Nowadays almost every mobile phone, every smartphone have included GPS receiver which could be used for various purposes. One of them is exactly Endomondo. This service is working on two levels: as web application and as app for mobile device. Method of working it is very easy. First, we have to create Endomondo account, optionally connect it with social networks, like Twitter, Facebook and finally download the app from our manufacturer's store. When you finally installed it, you should fill your profile up. It is necessary, because system will work out your training data, for example burning off calories, on base of this profile. After all this setting up you can open application, find GPS signal and start training. Now, your stop-watch will start counting time

and application will receive and save your coordinates from satellites. Moreover if you have access to the internet in your mobile device, Endomondo can use it for sending your geographical position to their databases. Furthermore it is happening in real-time, therefore everyone, who has got capabilities to see your profile and your trainings, can see, where are you now, on Google Maps overlay, like a spy. Unfortunately (or maybe exactly not?) we are living at *spy-times*. Endomondo does not bring so much data transfer, even when you are training a lot. Otherwise, if you do not want to be spied or have not got access to the internet, you can sync all your trainings with all details, at home, for example by wireless connection. On the computer, whenever and wherever you can check your achievements, compare it with others, join to the challenge, calculate how many hamburgers have you burnt or how many way to the Moon have you moved on. What does it give us? I think the most important thing is that it is bringing oneself to do more exercises, to run, to cycle, to walk. We feel internally mobilized to do that. Moreover, at home, we can check out where we have been. In addition Endomondo could be great tool for parents to control their children. For example we can create two accounts: for us and for a child. Then, on child's mobile phone (I suppose kid has got almost new mobile device with GPS antenna) we are configuring application. If our offspring wants to go somewhere with friends, but we want to know where are they, we can turn on Endomondo and track the child on a computer. I do not think it is spying, I think it is care for it, for its safety. If you are a professional sportsman or sportswoman you can also buy on Endomondo Store additional equipment like a Bluetooth heart rate monitor which will save current measurements on application statistics. It allows seeing how our heart was working during all training.

On the end of article I want to show I think the best web service, which is now using by more and more people. It is of course cloud storages – network file hostings. It began when speed of the internet quickly increased, average in 2005. Firstly, we had heard about RapidShare. This web service was giving about 100 megabytes storage per file. People were using it, shared photos from holidays, but the biggest disadvantage was the file had been deleted after 90 days if nobody downloaded it. Moreover only downloading process made some problems, because user who wanted to do it had to (and now they also have to) wait about one minute, when *downloading had been prepared*. Of course it was purposeful action made by RapidShare, because they wanted to pull in the biggest amount of users with bought account in

pro version. Accounts weren't free; therefore people who wanted to download immediately files had to pay for it. Meanwhile in Poland appeared completely revolutionary network drive. In Polish it was called *Chomikuj*, what in English means to hoard by hamster. It was metaphor to way of working it, to uploading a lot of files. This one was revolutionary, because it was wholly free for using and it was unlimited. People could upload everything what they want: photos, videos, backups, without paying attention to its size. But there was one snag: other users which wanted download our files had to have available transfer limit. Every week it is renewed up to 50 megabytes. If you want download more data, you should buy more transfer limit by SMS or money transfer. *Chomikuj* as first web drive shared their web client interface based on JavaScript and AJAX technology. Nowadays it is possibly to download special *Chomikuj* client for Windows, which allows downloading and uploading files to storage, even after troubles with internet connection. Similar service released Microsoft. They called it Windows Live Sky Drive. It is free as well and access to it user can get by the web browser. Microsoft gives 25 gigabytes storage and maximum 100 megabytes per one file. On a bit different rules works Dropbox. Dropbox is now surely the most known cloud web storage, which offers us up to 8 gigabytes free web drive. It differs than other because it is normally available from system's default files browser and of course from web too. Using Dropbox is very easy. We have to only create account and install client on computer. If we have mobile phone with new operating systems, like iOS, Android, we can install special application for it. Now, each files which we put onto Dropbox will be downloaded and saved on every device connect with your account. Unless you defined earlier that not to sync all folders at some devices. It is very easy – we are editing some document on our desktop. We send it to Dropbox and on a journey we can edit it in notebook. Not until notebook battery will be depleted, next we can continue editing document on the mobile phone. All this changes are saving on device's memory and in Dropbox cloud storage. Similar to Dropbox could be Box or Ubuntu One, but it is less well known than Dropbox. Finally I went to the last way to storage files in web cloud. It is brand new technology, brand new service, but I am sure in next months it will be more known. It is Bitcasa – infinite storage on your desktop. Sounds great! And it is great! For \$10 we will have unlimited cloud to upload files, but we cannot share this files with others, there is no option. Inventor's idea is all users' files should be on Bitcasa with no duplicates and syncing with hard drive as in Dropbox. On computer could be very

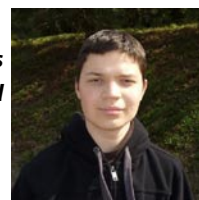
small hard drive, even 50 gigabytes, only for operating system and necessary programs, because all files will be on Bitcasa, which is seen by computer as unlimited external drive. Every time you want to get access your documents, spreadsheets, presentations, photos etc. you have to have connection to the internet. Without them you are alone with your computer and files only on it. It is disadvantage, but slogan tells – infinite storage in cloud, so it is not possible to sync infinite data on all the devices. I think that when internet connection speed will increase even more, Bitcasa could be really big competition, especially for Dropbox. Providers of cloud storages services have problems with pirates. But some of them allowed them, because they are making money on it. For example few months ago FBI closed *Megaupload.com* and arrested their owners under the charges of criminal copyright infringement in relation to his service. RapidShare had also problems with copyrights. Now they changed their policy and piracy files are immediately deleted. Of course if you are cloudifying legal files, which you are an owner you can sleep well, because all of them are safe in a lot copies on storages in the all the world!

Nowadays, we cannot imagine world without the internet. We cannot imagine world without web applications and services. It is part of our life. With time it becomes more useful, more practical and more mobile. It is not a secret that we are using many more mobile devices than ever before. This market is still growing up, therefore we can expect even more. Soon majority of web applications will be made in more effective technology, which now there is. I mean HTML 5. But what about safety our data, our money and ourselves? I think if we can manage all this services reasonable with a bit of vigilance we can sleep well. ♦

JAN POGOCKI

17-years-old computer geek who likes testing new products, write tutorials and his technological blog.

Website: www.janpogocki.pl





Global I.T. Security Training & Consulting

www.mile2.com

In February 2002, Mile2 was established in response to the critical need for an international team of IT security training experts to mitigate threats to national and corporate security far beyond USA borders in the aftermath of 9/11.



IS YOUR NETWORK SECURE?



A Network breach...
Could cost your Job!

Available Training Formats

1. F2F Classroom Based Training
2. CBT Self Paced CBT
3. LOT Live Online Training
4. KIT Study Kits & Exams
5. LHE Live Hacking Labs (War-Room)

Other New Courses!!

- ITIL Foundations v.3 & v.4
- CompTIA Security+, Network+ & CISSP & CAP
- SANS GSLC GIAC Sec. Leadership Course
- SANS 440 Top 20 Security Controls
- SANS GCIH GIAC Cert Incident Handler

Worldwide Locations



CISSP™
C)ISSO
C)SLO
ISCAP

GENERAL SECURITY TRAINING
CISSP & Exam Prep
Certified Information Systems Security Officer
Certified Security Leadership Officer
Info. Sys. Certification & Accred. Professional



C)PTE™
C)PTC™

PENETRATION TESTING (AKA ETHICAL HACKING)
Certified Penetration Testing Engineer
Certified Penetration Testing Consultant



C)SCE™

SECURE CODING TRAINING
Certified Secure Coding Engineer



C)WSE™
C)WNA/P™

WIRELESS SECURITY TRAINING
Certified Wireless Security Engineer
Certified Wireless Network Associate / Professional



DR/BCP

DR&BCP TRAINING
Disaster Recovery & Business Continuity Planning



C)SVME™

VIRTUALIZATION BEST PRACTICES
Certified Secure Virtual Machine Engineer



C)DFE™

DIGITAL FORENSICS
Certified Digital Forensics Examiner



We practice what we teach....

Other Mile2 services available Globally:

1. Penetration Testing
2. Vulnerability Assessments
3. Forensics Analysis & Expert Witnesses
4. PCI Compliance
5. Disaster Recovery & Business Continuity

(ISC)2 & CISSP are service marks of the IISCCC. Inc. Security+ is a trade mark of CompTIA. ITIL is a trade mark of OGC. GSLC & GCIH are trademarks of GIAC.

1-800-81-MILE2
+1-813-920-6799
11928 Sheldon Rd Tampa, FL 33626

Web Services and Testing

The articles listed below provide an overview about web-services and its testing. The main purpose of this article is to give an overview in testing web services. This article would be helpful for testers, developers, Project Managers who does not have more technically strong in web services field. Service-oriented architecture (SOA) and web services are very popular topics in many development projects.

Web Services can convert your application into a Web-application, which can publish its function or message to the rest of the world. The basic Web Services platform is XML + HTTP.

A web service is just a system which resides somewhere on a network and gives response specific requests from clients. Here you must be aware of the term clients, clients are not only the people working on their computers on the desk, and it can also be some machines.

Web Services are just a simple combination of different technologies that allow for making connections between different computers, machines, people over internet. Here you must be aware about why we are using the term *Services*, so *web is just a term or another*

name for internet but services provides us connectivity with other users over internet or web.

A Web service is a communication service or method through which machines (Computers, Mobiles, Notebooks, and PDA etc) can communicate over the web (internet). It is a great step towards simple access to software and data over the network; it will be easier if you see the Figure 1 which I have sketched out here.

Here our web client is looking for a web page *www.google.com* as you can see in our example, so just open your browser and put the URL and hit the Enter button, now this request is received by Google web server and it sends the *Google webpage* to your browser, you can skip all the deeper functionalities about how it works. So the basics are simple.



Figure 1. Interaction between web client and google web server

And there is nothing new in the world of web-servers over the Internet:

- The web service provider provides you some data, information, software etc.
- A computer off course (browser+computer) makes a request for the web services across the network
- The web service performs some action, and sends the response back to you

Now we must take a look at *Testing of web services*.

There are lots of different components in a complete Web service, so one should be not surprised about this saying that complete testing is difficult. There are different technologies involved in even a simple client and server. Network part, server-side code, database, html, CSS are also part of it so testing is not an easy task, so we must keep some basic principles of testing in our mind. So take a look at some web-service testing principles according to my opinion.

Don't forget to test

Testing is an important part of every development project. Web service is an important part of any development project so it is very important to perform it in the initial stages of your project.

Challenges in testing web services?

It is very necessary to test web-services; the main reason behind this is that websites are the most visible component on the internet so it must be very robust.

There are two types or categories of web-services, first is web services running in intranet environment and other is web-services running on the internet.

Intranet web-services provide web facilities only to the people within a particular organization for example office-automation facility, in which employee of that organization update his personal information, he can view his salary, his debits about his pension.

In case of internet web-services the concept is little bit different, this service is for whole world not for a particular group, here whole world can view the website and technically speaking this web-service runs on any PUBLIC IP Address. Any one can view and use this service.

So testing intranet web-services and internet web-services are different in terms of security and scalability. Intranet web-services are used only by internal users (users within an organization) but internet web-services are used by whole world and anyone one can access it, so scalability and security must be the main consideration here.

There is one more challenge in testing web-services, they are completely UI-less means there is not any graphical user interface that can be tested, it is very hard to test it manually. So for testing web-services one must know some programming skills.

Why Testing is Needed

There are lots of reasons of testing web services and the main one is vulnerability, it is everywhere on the internet as well as on application servers and hackers and bad elements always try it out to crack it break it hack it for stealing valuable information details such as your bank account number, online banking passwords, income details etc etc... so keep it safe boy. Before you make all your web services available to the public, however, you need to make sure they work. The only way to do this is by writing functional tests for your web services.

Listing 1. Returned results

```
import groovy.net.soap.SoapClient
def proxy = new SoapClient("http://localhost:8080/MathToolInterface?wsdl")
def result = proxy.add(5.0, 2.0)
assert (result == 7.0)
result = proxy.square(5.0)
assert (result == 25.0)
import groovy.net.soap.SoapClient
def proxy = new SoapClient("http://localhost:8080/MathToolInterface?wsdl")
def result = proxy.add(4.0, 6.0)
assert (result == 10.0)
result = proxy.square(6.0)
assert (result == 36.0)
```

Some technical terms

So now we shall look further at some technical details about web services.

Most of the web services around the world are running on unix-linux-os based computers, so we can easily change a computer to a web-server with some easy configurations.

So I am just assuming you are using fedora Linux operating system, and now we will need some software stuff to create or change a normal desktop computer to a normal web-server

For this you need a software package named Apache Tomcat and run a service httpd on your web server.... now test it by typing `http://localhost` on your web browser, normally web service runs on port 80 and 8080.

Types of testing

There are different types of testing which can be performed on web-services.

- **Functional testing:** Here one must test the functionality of web-services, here we test different inputs and observe the outputs and consider that everything is correct or not? Does this web-service supports all protocols? Does this web-service support security and authentication? This is because one has no control on the web-service clients. They are independent to request to web-server.
- **Load/scalability testing:** Here load testing means how web-service handles the request of users in terms of scalability. Suppose your web-service is working very fine when only one user is requesting to web-server, now the main question is that what should happen when the number of users increased means whether the web-service will work smoothly when the number of users increased. So this is also known as scalability test because here scale just defines the number of users requesting for web-service.
- **Regression testing:** This test is just a cut-down version of a functional testing. Here one must check that whether web-service is still working when there is a change in build, version and releases of the software. Let's assume that your programming team has changed the coding style or method of addition, subtraction. Now regression testing checks whether web-service is still working in this situation, is it addition and subtraction works well and the output is correct as before.
- **Web-service monitoring:** When your web-service is in running position and used by clients, now

it is the right time to monitor your web-services continuously. Here one should monitor the request time, response time, load on the web-service, web-service busy time.

The main purpose of this testing is to tell you that whether you are on the right way and right direction. So here we focus on the initial phase and one has to made choice in this testing like which database, which programming language, which testing tool should be used. These issues must be resolved before starting means early in the development life cycle; this will save your money and time. You must also know how the clients will be accessing the web services.

Principles of testing

Let's review some of the general principles of testing and debugging, it does not depend on what tools and coding styles you are using.

- We should keep it in our mind that Inputs and Outputs are the most important components in

Listing 2. Traditional web test



```
<steps>
  <invoke method="POST" contentFile="add.xml"
          soapAction=""
          url="http://localhost:8080/MathInterface" />
  <verifyXPath xpath="//addResponse/out[
                text()='3.0' ]"/>
  <invoke method="POST" contentFile="square.xml"
          soapAction=""
          url="http://localhost:8080/MathInterface"/>
  <verifyXPath xpath="//squareResponse/out[
                text()='9.0' ]"/>
</steps>
<steps>
  <invoke method="POST" contentFile="add.xml"
          soapAction=""
          url="http://localhost:8080/MathInterface"/>
  <verifyXPath xpath="//addResponse/out[
                text()='3.0' ]"/>
  <invoke method="POST" contentFile="square.xml"
          soapAction=""
          url="http://localhost:8080/MathInterface"/>
  <verifyXPath xpath="//squareResponse/out[
                text()='9.0' ]"/>
</steps>
```


web-services. There must be a proper choice for inputs and outputs, it will allow testing without any other HTTP server or Network related complexity or problems.

- Everything has parts so it will be more convenient for a user to test separate parts in all situations. Unit testing is also preferred by many developers and programmers. One must use language functionalities such as assertions in Java to catch and identify bad inputs to functions.
- One must use understandable and explanatory names of interfaces, variables, classes, functions, methods, files so that everyone can easily understand the code written by other one.
- Documentation is also very necessary part of it so one should do this always during testing.

There are millions of books, articles, technical papers, documents, seminars on the design and coding of Web services and *service oriented architectures* (SOA).

It is very necessary to know about testing, deployment and management of Web services-based architectures.

With Web services, standards for data format (XML), communication (SOAP) and programmatic interface (WSDL) are data-driven, simple and share a common XML-based foundation. There are lots of traditional tools for testing and it is also important to validate interface points and message formats rather than simply testing at the graphical user interface level.

We can use different tools for example Groovy, Web Test and SoapUI.

Web Services can be tested in several ways. Some are following.

- Work like a normal web services client and perform asserts on the returned result
- By using Web Test (Groovy syntax or with either the XML)
- By using SoapUI (functional and load testing)

Listing 3. ad.xml

```

<?xml version='1.0' encoding='UTF-8'?>
  <soap:Body>
    <add xmlns="http://DefaultNamespace">
      <in0 xmlns="http://DefaultNamespace">1.0</in0>
      <in1>2.0</in1>
    </add>
  </soap:Body>
<?xml version='1.0' encoding='UTF-8'?>
  <soap:Body>
    <add xmlns="http://DefaultNamespace">
      <in0 xmlns="http://DefaultNamespace">1.0</
        in0>
      <in1>2.0</in1>
    </add>
  </soap:Body>

```

Listing 4. square.xml

```

<?xml version='1.0' encoding='UTF-8'?>
  <soap:Body>
    <square xmlns="http://DefaultNamespace">
      <in0 xmlns="http://DefaultNamespace">3.0</in0>
    </square>
  </soap:Body>
<?xml version='1.0' encoding='UTF-8'?>
  <soap:Body>
    <square xmlns="http://DefaultNamespace">
      <in0 xmlns="http://DefaultNamespace">3.0</in0>

```

```

  </square>
</soap:Body>

```

Listing 5. Groovy within webtest

```

<Steps>
  <groovy>
    import groovy.net.soap.SoapClient
    def proxy = new SoapClient("http://localhost:
      8080/MathInterface?wsdl")
    def result = proxy.add(3.0, 8.0)
    assert (result == 11.0)
    result = proxy.square(7.0)
    assert (result == 49.0)
  </groovy>
</steps>
<steps>
  <groovy>
    import groovy.net.soap.SoapClient
    def proxy = new SoapClient("http://localhost:
      8080/MathInterface?wsdl")
    def result = proxy.add(6.0, 6.0)
    assert (result == 12.0)
    result = proxy.square(9.0)
    assert (result == 81.0)
  </groovy>
</steps>

```

Being a normal web service client

Here one can be a normal web service client and perform asserts on the returned results: Listing 1.

Using Web Test

It is just a combination of your tests into an acceptance test suite. It is just a traditional Web Test: Listing 2. Here *add.xml* is following: Listing 3 and *square.xml* would look something like: Listing 4. Here we are using Groovy within webtest, have a look on it: Listing 5.

Note: Its time to place the jars in web test lib directory.

Using SOAPUI

SOAPUI is the most popular tool in the field of web testing. It provides a GUI (*Graphical user interface environment*). It is an open source web service testing tool for service-oriented architectures (SOA) and it also provides free and open source cross-platform Functional Testing solution. Its functionality covers web service inspection, invoking, development, simulation and mocking, functional testing, load and compliance testing. It provides a Web service client that can automatically generate Web service requests and tests (Figure 2).

It is a SOAP functional and load testing tool. It use Groovy steps within its test cases. The soapUI is designed to simplify the testing of your Web services; It is very useful

for interacting with third-party Web services and one can easily expect the response with the help of this tool.

SoapUI supports multiple protocols such as SOAP, REST, HTTP, JMS, AMF and JDBC. It enables you to create advanced Performance Tests very quickly and run Automated Functional Tests. It is easy to use for both technical and non-technical person.

Now take a look at Different TOOLS for web-service testing

- **Ranorex:** It is a Windows GUI test automation framework. It supports the testing of many different applications like Web 2.0 applications, Win32, MFC, WPF, Flash/Flex and .NET (Figure 3). Ranorex doesn't have a scripting language of its own. A user or you can say that a tester can use different languages like C#(C Sharp) and VB (Visual Basic) as its base; it is easy to accomplish this because of GUI environment provided by Ranorex.
- **Selenium:** It is a portable software testing framework for web applications. There is no need to learn any scripting language (Selenium IDE); it provides a very efficient concept as in Music Players, just record and then playback. It also supports a test domain-specific language (Selenese); It is used to write tests in different popular programming languages like C#,

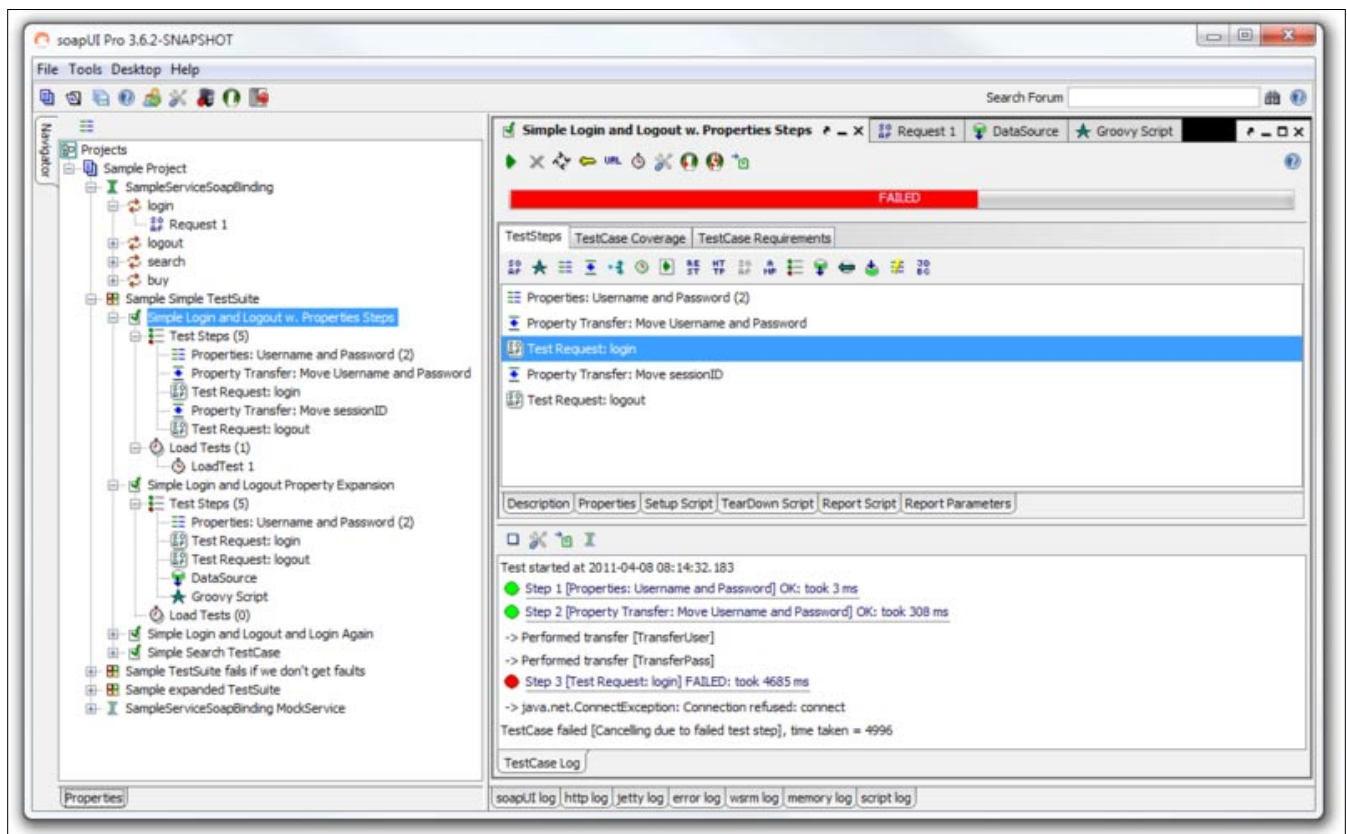


Figure 2. Web services requests and tests

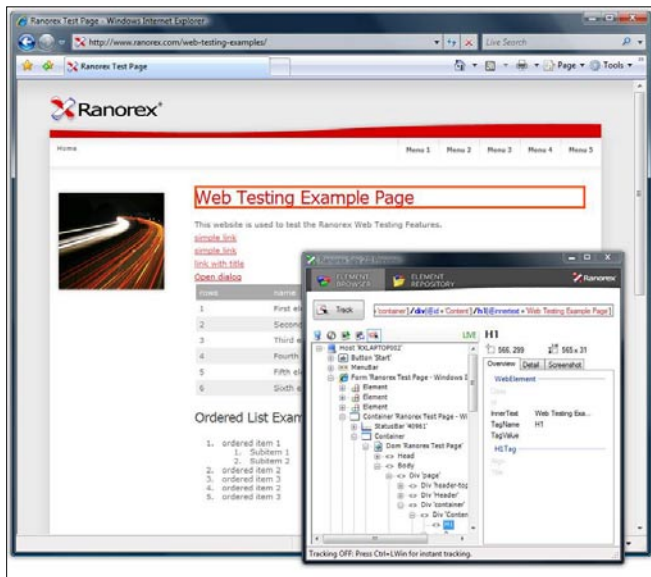


Figure 3. Windows GUI test

Java, Perl, Groovy, PHP, Python and Ruby. Now one can run these tests on different browsers like Mozilla, Google Chrome, and Opera etc. It can run on different platforms such as Windows, Macintosh, and Linux. It has following features so take a summarize look at them.

Features

- Record and playback
- Intelligent field selection will use IDs, names, according to its need.

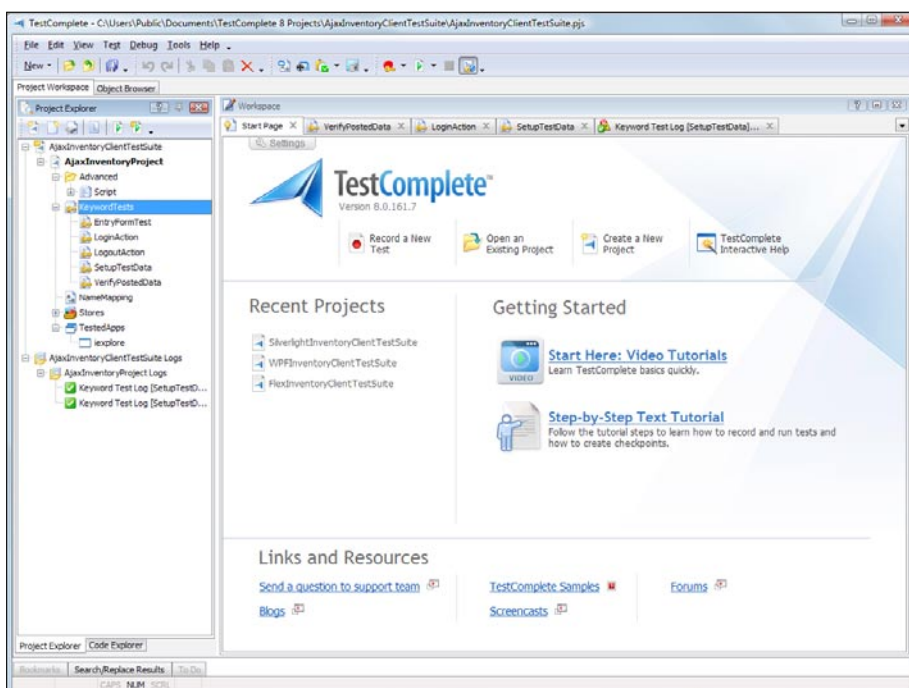


Figure 4. Automat ed testing tool

- It supports Walk through tests.
- Debugging is provided and one can set break-points.
- Tests can be saved in different formats such as Scripts like Ruby.
- .js extension is supported by it.
- Assert title of pages automatically.
- *Soatest: Parasoft SOAtest* is a very nice comprehensive in nature and it also works as an analysis tool, and very better for the Service Oriented Architectures which needs validation. With the help of this tool a tester can easily test the functionality of their services at message layer and it has one more feature that many of the transport protocols are being supported by it such as TIBCO, .NET WCF HTTP, .NET WCF TCP, HTTP 1.0, MQ, RMI, HTTP/1.1, JMS, SMTP. It provides testing on the different levels or you can categorize it in following types unit testing, security testing, regression testing, static analysis, and load testing. Many companies and organization such as Siemens, Medic Alert, and AOL use this tool.

Test Complete: It is an automated testing tool, developed by SmartBear Software. Its main concept is to allow testers to create software quality tests. One can record tests, tests can be manually scripted as well as created manually with keywords, and it also supports error logging. Tests can be recorded, manually scripted or created manually with keyword operations and used for automated playback and error logging (Figure 4).

Test Complete is used for testing many different application types including Web, Windows, and WPF, Flash, Flex, Silver light, .NET and Java. Record and playback test creation records a tester performing a manual test and allows it to be played back over and over again as an automated test. It automates front end UI/functional testing and back-end testing like database, and HTTP load testing. Recorded tests can be modified later by testers to create new tests or enhance existing tests with more use cases. It has following features.

- **Keyword Testing:** It has already an embedded built-in keyword-driven test editor that consists of keyword operations.
- **Issue-Tracking Support:** It has built-in issue-tracking templates; with this one can easily modify, create and delete issue-tracking items stored in issue-tracking systems.
- **Test Visualizer:** It can capture screenshots automatically during test recording and playback. This enables quick comparisons between expected and actual screens during test.
- **Support for plugins:** It also supports the integration with third-party applications because of built-in plugins facility .Other parties can connect their application with it.
- **Full-Featured Script Editor:** It has already an embedded built-in code editor with a set of special plug-ins that helps testers write scripts manually.
- **Test Record and Playback:** It supports record and playback feature, so one can easily replay his action and undo what he wants.
- **Script Debugging Features:** it also provides the feature of debugging the script written by coder so one can easily debug the script.
- **Access to Methods and Properties of Internal Objects:** It is very efficient in reading the internal and shown elements of applications like Delphi,

C++Builder, .NET, WPF, Java and Visual Basic applications, it also allows test scripts to access these values for verification or use in tests.

- **Unicode Support:** One can easily test a non-ASCII application which uses Unicode character sets, it supports Unicode character set.

Tosca

TOSCA Test suite is a software tool for the automated execution of functional and regression software testing. It also has following features

- A graphical user interface (GUI), a command line interface (CLI).
- Application programming interface (API).
- It supports the test automation functions.
- TOSCA includes integrated test management.

TOSCA is a test management, design, execution and data generation toolset for functional and regression tests. TOSCA Test suite consists of the following:

- TOSCA Commander, it is the test suite's execution tool which is used to create, administers, execute and analyze test cases.
- TOSCA Wizard, stores the technical information XML-GUI Maps called modules for creating a model of the application.

- Once test cases have been created, TOSCA Executor executes the test cases and displays the results in TOSCA Commander.

- TOSCA Exchange Portal, a portal where customers can use and exchange special modules, extensions and prebuilt TOSCA Commander components (subsets).

- The Test Repository, which includes integrated version control, stores all test assets and can be accessed by multiple users.

Watir

Web Application Testing in Ruby (or *Watir*, pronounced *water*) is a toolkit used to automate browser-based tests during web application development. This automated test tool uses the Ruby programming language to drive Internet Explorer,

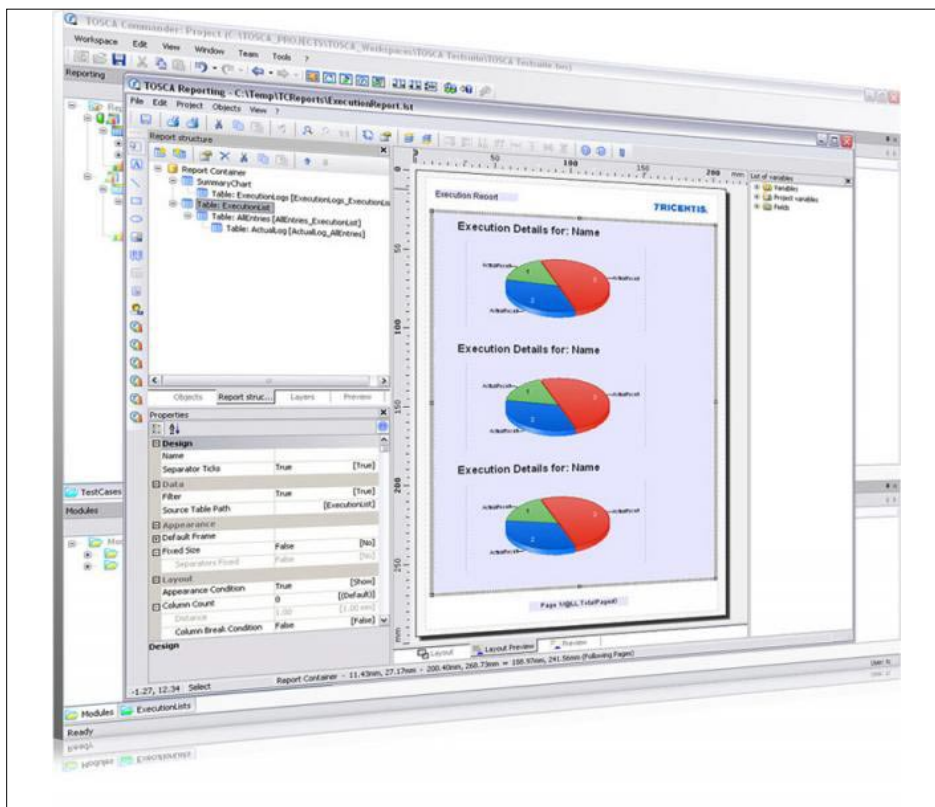


Figure 5. Watir

Mozilla Firefox, Google Chrome, Opera and Safari, and is available as a Ruby Gems gem.

Watir, pronounced water, is an open-source (BSD) family of Ruby libraries for automating web browsers. It is simple and flexible. Watir is a family of Ruby libraries but it supports your applications. . It allows you to write tests that are easy to read and maintain. Ruby gives you the power to connect to databases, read data files and spreadsheets, export XML, and structure your code as reusable libraries. Unlike other programming languages, Ruby is concise.

It uses the browsers in the same manner as people or human do. It clicks links, fills in forms, presses buttons, type data etc. It also checks the results.

Why Watir?

- It supports multiple browsers on different platforms.
- It is powerful and easy to use, yet beautifully lightweight.
- It's a free Open Source tool. There are no costs to use the tool.

Listing 6. Request to a web-service

```
<env:Envelope
xmlns:env="http://www.w3.org/2012/03/soap-envelope">
<env:Body>
  <m:ValidatePostcode
env:encodingStyle="http://www.w3.org/2012/03/soap-
encoding"
xmlns:m="http://www.abcd.com/Postcode">
<Postcode>208006</Postcode>
<Country>IN</Country>
  </m:ValidatePostcode>
</env:Body>
</env:Envelope>
```

Listing 7. Response to the request

```
<env:Envelope
xmlns:env="http://www.w3.org/2012/03/soap-envelope" >
<env:Body>
  <m:ValidatePostcodeResponse
env:encodingStyle="http://www.w3.org/2012/03/soap-
encoding"
xmlns:m="http://www.abcd.com/Postcode">
<Valid>Yes</Valid>
  </m:ValidatePostcodeResponse>
</env:Body>
</env:Envelope>
```

- It supports your web app no matter what it is developed in.
- There's a very active and growing community behind it.
- It uses Ruby, a full-featured modern scripting language, rather than a proprietary vendor script

So now we have already seen some tools which are most used for web-services testing; now we will talk about some standards here.

Standards

We use many open protocols for calling services and transmitting data. web-services use these protocols.

In past web service providers have their own data formats and standards but after the existence of XML everything has gone changed, now one can rely on simple extensible Markup Language (XML) and it is based on simple plain HTTP protocol. It provides some easier access to web-pages and request-response is easier.

The Simple Object Access Protocol (SOAP) is a W3C (world Wide Web Consortium) standard protocol that defines the format for web service requests.

SOAP messages are sent back and forth between the service provider and service user in SOAP envelopes, here SOAP envelopes just contains a request for an action and it also store the results of that action. SOAP envelopes are XML formatted, and one can easily decode it. Now take a look at simple SOAP request and this is sent via an HTTP request to a web-service: Listing 6.

The key elements of our SOAP envelope are easy to recognize: two parameters (postcode and country) are contained within an element named Validate Postcode, which happens to be the name of the web service we are calling. Other data within the envelope, like the text encoding and SOAP version, helps the web service process the request. A response to this request might look like this: Listing 7.

Here you can easily interpret this message, here ValidatePostcodeResponse element is just validating or you can say that answering the request generated by ValidatePostcode.

Please read this article and send me acknowledgement regarding your views @ saurabh@iitk.ac.in. ♦

SAURABH MALHOTRA

Junior Tech. Dept. of Computer Science and Engineering
Indian Institute of Technology Kanpur Kanpur-208016, INDIA
„This article is dedicated to my parents and Great Giani Sant Singh Maskeen“.

Basic Do-It-Yourself

Website SEO Audit & Optimization for Search Engines

Building a completely search engine optimized website from scratch on an SEO friendly structure may seem an easier task than bringing an old one to work better with search engines.

However, it is never too late to effectively optimize a website design for SEO to enjoy higher visibility and rankings, improve site's credibility, get pages indexed smoothly and stop missing out on traffic. Follow our basic SEO audit and optimization guide to make the most of search engines algorithms and achieve better results for your website in SERPs.

Flash

Stay away from Flash as search engine robots don't index Flash files, which means that all the valuable text content inside will be ignored by search engines. Using Flash for the entire site will kill your SEO efforts unless you build a separate Flash-free website to enable internet users find your site in the search engine results pages. An immediate makeshift solution for completely Flash-designed sites is to put, inside of a website (not just as an option), a link to view the website in Flash. If you can't do without some Flashy enhancements, then adding an alternative textual description for Flash file is an absolute must.

Frames

Using frames is equally bad for SEO, making it difficult for search engine robots to crawl websites. Most of search engine crawlers get such error message

while trying to visit a frames-built site: *You need a frames-browser to view this site.* The web content inside frames will not be indexed by search engines that don't support frames. A quick way out is to add the `<noframes>` tag to have some of the content indexed. Google crawls frames, however, to the extent it can, so you may be getting poorer SEO results. It officially states:

Google supports frames to the extent that it can. Frames can cause problems for search engines because they don't correspond to the conceptual model of the web. In this model, one page displays only one URL. Pages that use frames display several URLs (one for each frame) within a single page. If Google determines that a user's query matches the page as a whole, it will return the entire frame set. However, if the user's query matches an individual frame within the larger frame set, Google returns only the relevant frame. In this case, the entire frame set of the page will not appear.

Source: Google Webmaster Help Center

CSS Navigation

Using a *cascading style sheet* (CSS) navigation menu is good for SEO. It helps search engine robots to easily crawl and index navigational text. Flash or Javascript navigational menu makes it impossible for crawlers to index or follow the links. Incorporating a multi-level

The article was provided by Profesjonalne [Pozycjonowanie](#), a leading SEO and PPC company based in Poznań.

CSS drop down navigational menu brings even better results as it enables to incorporate each and every page of a website into navigation menu. CSS drop down navigational menus stand for top usability, instant access to web content and most positive experience with website. Visitors will need many fewer clicks to get to any page they want and search engines will find web pages faster. CSS drop down navigational menu creates a sitemap of all the links available to search engine robots.

CSS Stylesheets

CSS stylesheets improve loading times through reducing the amount of code on web pages. Fast loading pages allow search engine robots to crawl them much quicker. CSS stylesheets, unlike tables, enable great control as well as quick and easy changes to all website design elements like heading tags (h1, h2, h3 etc.), paragraphs, divs, navigation, images, links.

Keywords in Title Tag

Title tag of a website and its pages, including your keywords, is crucial for SEO. Title tags show in the search engine results pages for your website. Keep title tags for all the subpages different, user-friendly and short (some 6 to 8 words, up to 65 characters). Don't repeat the same keyword in your title tag over and over again – to search engines it may look like an attempt to spam or manipulate SERPs. Put the most important and relevant search term at or near the start of the tag to communicate to search engines what keywords the website and its web pages should be indexed for.

Code sample:

```
<head>
<title>This is an example of a title tag</title>
</head>
```

Meta Description Tag

Provide unique meta description tags for web pages, i.e. compact and to the point explanations of the web contents. Try to smartly incorporate your keywords. Optimum description length is c.a. 160 characters. Descriptions are commonly used by search engines on search result pages to show preview information for a particular page. Description tags are absolutely vital from the point of view of searchers letting them instantly know what they can expect to find and encouraging them (or not) to click. Good description will get you high user click-through from SERPs.

Code sample:

```
<head>
<meta name="description" content="This is an example of
a meta description.">
</head>
```

Keywords in URLs

Incorporating keywords into the URLs of web pages and website's folders is also a good SEO-friendly idea. To best include several phrases in a filename, separate them with dashes '-', e.g.: 'yourwebsite.pl/profesjonalne-pozycjonowanie-stron-www/'.

Keyword Density

Use keywords that are already incorporated into title tags, URLs, file names, etc. also in text content of corresponding web pages. Again, remember to avoid packing keywords in your web content as excessive repetition could harm your SEO job. Check out your keyword density with one of free keyword density tools that are available. Recommended keyword density ratio in the body text is in the range of 2% – 8%.

Anchor Text & Keywords

To make a website work better with search engines and become more user-friendly, put keywords in the anchor text of hyperlinks. If a given page is dedicated to laser liposuction, then the hyperlink's anchor text should be: Laser Liposuction. With that in place, you inform search engine robots what this page is about.

Keywords & Headings

Use your keywords in the heading tags (h1, h2, h3) of web pages to enhance their significance and weight. Never repeat the same keywords in one heading tag.

Images

Place 'Alt' tags for all images you have on website to tell search engine robots what each image is about and try using your keywords in the alt tags. Don't put text content inside images because search engines won't be able to crawl or index your text information then. It will also add to picture size increasing download times.

MONIKA BAŃCZEROWSKA

The article was provided by Profesjonalne [Pozycjonowanie](#), a leading SEO and PPC company based in Poznań.

Interview with

Tom Brennan



Tom's colossal cave adventure started the same year as WarGames armed with a Televideo 802H, Commodore and Atari 8-Bit machines and a set of lock-picks the hobby moved quickly from handles to mainstream. Tom took a front row seat on the architecture, development, administration and security of computer-controlled systems with experiences ranging from the financial trading floor of Wall Street to the United States Marines Corps.

In my day job I am surrounded by hundreds of subject matter experts at Trustwave. As the Director of Strategic Initiatives, at Trustwave SpiderLabs we service clients as the largest red team in the world focused on response and investigation, analysis and testing, research and development. Trustwave has over 1000 employees and is headquartered in the United States in Chicago, Ill. with offices throughout Africa, Asia, Australia, Europe, North America and South America.

For OWASP Foundation I volunteer my time to the community as a project contributor, project leader, NYC Metro and New Jersey chapter leader from 2004-current and the OWASP International Board of Directors 2007-Current.

For readers who are new to OWASP, please tell us more about the organization, it's goals and size.

TB: *The Open Web Application Security Project (OWASP)* is a 501(c)(3) not-for-profit worldwide charitable organization focused on improving the security of application software. Our mission is to make application security visible, so that people and organizations can make informed decisions about true

application security risks. Everyone is free to participate in OWASP and all of our materials are available under a free and open software license. There are over 150 projects at OWASP and over 160 chapters around the world today and over 30,000 participants and growing daily.

Does OWASP collaborate with other security-related organizations?

TB: Yes, OWASP Foundation collaborates with many other organizations globally. With over 30,000 participants OWASP has impacted both compliance and standards globally. There are over 30 citations to OWASP Foundation including: *Payment Card Industry (PCI)*, *Centre for the Protection of National Infrastructure (CPNI)*, *Cloud Security Alliance (CSA)*, *European Network and Information Security Agency (ENISA)*, *National Institute of Standards and Technology (NIST)*, *Information-Technology Promotion Agency (IPA)* in Japan just to name a few.

Reference URL: <https://www.owasp.org/index.php/Industry:Citations>.

How is OWASP helping promote Web Application Security?

TB: Everything we do goes back to the mission of and core values of the organization. OWASP is a platform for the industry of software and its security. As an example; we provide a global ecosystem for individuals to start a new project or contribute to any of the existing ones. We provide global and regional conferences to bring the best and brightest in the world together do discuss building, breaking and defending of software security in a responsible manner.

Does OWASP have any plans of starting certification program?

TB: OWASP will continue to provide free and open security resources. Others may choose to use this material as a foundation for a certification programs, but we will never restrict access to our material. OWASP is not a certification body. The organization is run by consensus of its membership, as an example the idea of a Certification Program was one of our topics during the 2008 Summit and again during our 2010 OWASP Summit (OWASP Summits bring together our leaders for a frank face-to-face discussion that sets the tone and direction of the organization). Although there were advocates for both sides, the result was for OWASP to stay pure to its mission and not become a certification business. It was decided to leave that venture to those that wish to develop, manage certifications and as always our materials are open to be used as reference and study material.

OWASP Top 10 risk is very popular in the industry but it seems a little high-level.

Where can we find more details information about the risks, mitigation techniques and current state.

TB: The OWASP Top 10 (https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project) is clearly one of the

best known documents produced by project leader Dave Wichers, Jeff Williams and The Top 10 is a simple way to communicate security to end users and is very effective. This project has been translated from English to now include Italian, Chinese, Spanish. Concerning it being *high-level* it is and by design when experts were polled for information concerning the 10 most important risks in application security based on over 49 classes of attack types this was the resulting information. After digesting the OWASP Top 10 we encourage interested people to learn more online at www.owasp.org reviewing our other release quality projects as well as those in beta and alpha stage. For your readers looking to understand how to those looking to understand how to perform layer 7 application security penetration testing, I would recommend they read the OWASP Testing Guide. This how-to book is FREE and online had contributors from around the world on the manual techniques used to identify classes of attack. It should also be noted that finding flaws is easy when you know where to look, defending or building security into software is just as important but to the media as example and for some new folks to the professional building security is not as sexy as breaking but I would say more important.

How much focus is on security education? Do you have tutorials, trainings and videos accessible to the members or public-at-large?

TB: Indeed we do. Recently training videos were put online by contributors that include Jerry Hoff, Nishi Kumar, Keith Turpin, Kauai Hinojosa and others. We also have been video taping our Application Security Conferences and putting the videos online from them of the speakers at no-charge, simply a community you can find them online. We also offer technical training at our conferences and regionally Education is key to improving security, first you have to understand the issue then collectively we can improve it together. It is important to understand that OWASP is like public radio and a collective of volunteers globally. All of our materials is free and available for anyone to use. We do have membership available for individuals and corporate supporters for a donation to allow us to continue the mission. OWASP does have a small staff of employees (Kate, Sarah, Kelly, Alison) that handle the back-office and operational daily logistics but the core is volunteers including the roles of global committee members, chapter leaders, project leaders and the international board of directors.

What are some of the exciting projects currently brewing at OWASP?

TB: There are many projects in motion globally over 30 new ones that are coming online (<https://>

www.owasp.org/index.php/Category:OWASP_Project). OWASP ZAP Project, OWASP Mobile, OWASP WAF Project with Mod_Security and the OWASP Application Security Assessment Standards Project are the ones I am following this month. At OWASP we also expect many improvements to existing projects and new ones to come online as OWASP has been accepted into the Google Summer of Code 2012 (<http://www.google-melange.com/gsoc/homepage/google/gsoc2012>) effort, this is exciting for OWASP to be part of.

What companies are associated with OWASP and how do they benefit?

TB: Today there are over 90 global supporters listed on the OWASP website and we appreciate every one of them. There was a study that asked why do organizations allow their people to participate or support non-profit groups, the survey revealed the following information: 94% think volunteering adds to the skills of their workforce. 58% say voluntary work can be more valuable than experience gained in paid employment. 25% offer paid time off to employee volunteers. 15% allow sabbaticals for volunteering projects. Employer supported volunteering can help a company's: Reputation and credibility, Recruitment and staff retention, Staff morale and work performance, Training and development, Change management, Government and regulatory relations. I think that summarizes it pretty good for most of our supporters. So as a professional community we provide a vendor neutral collaborative platform for this community and that also provides common ground for personal relationships of common interests and this leads to careers inquires to the active contributing organizations to our projects.

How can people get involved in OWASP activities?

TB: That is easy, attend one of the Global AppSec events. Join a project mailing list and say *Hello World* from the comfort of your own home. If you want to venture outside, submit a talk or come hear presentations from software security peers from around the world or meet-up with the local chapter members in your region.

Does OWASP provide any support or guidance to professionals who work in the regulatory and compliance industry (HIPAA, PCI, GLBA etc)?

TB: Yes. Those compliance frameworks already reference OWASP Foundation so I would start with our citations page to clarify context see: (<https://www.owasp.org/index.php/Industry:Citations>) There are thousands of individuals that day-to-day work in these areas and those connections are made by joining

chapter mailing lists and attending a local chapter event or conference.

A lot of our readers are located worldwide, do you have local chapters outside USA?

TB: Yes, OWASP Foundation is global. We are a US Based 501(3)c non-profit and we recently became a recognized organization in the European Union and France as example. With conferences being held in 2012 in Austin Texas, Athens Greece, Sydney Australia, Buenos Aires, Argentina and a regional event in Gurgaon, Delhi India this year just to name a few (https://www.owasp.org/index.php/Category:OWASP_AppSec_Conference) With over 160 chapters around the world (https://www.owasp.org/index.php/OWASP_Chapter) there is a local group meeting regularly and if there is not one in your region already starting a new one is easy.

Any conference recommendations related to Web Application Security and other information resources such as blogs, books and website.

TB: From a software security perspective OWASP conferences are clearly a focused group with the best and brightest minds in the world. What makes our events unique is that topics are technical, events are organized by local teams in that region with the assistance of the global conferences committee and our staff. For other information security resources I would encourage people to look for important news from OWASP on our blog at <http://owasp.blogspot.com/> note that anyone anyone is welcomed to contribute content. There is also Twitter, Facebook, LinkedIn social media groups. For other information resources to help with this onslaught of application security news, the OWASP team reviews over 130 of these sources and produces the OWASP Moderated Application Security News Feed and this can be found at: https://www.owasp.org/index.php/Application_Security_News.

Who is on your Information Security Celebrity list.

TB: We have a lot of software security rockstars at OWASP Foundation that contribute tons of time and energy. Some of these folks serve on the OWASP Global Committees, run projects or are very active chapter leaders. In addition to them there are 90 or so interviews with information security leaders at the OWASP Podcast that is hosted by Jim Manico, Exotic Liability Podcast with Ryan Jones and Chris Nickerson and SilverBullet Podcast with Gary McGraw. ♦

by Aby Rao

Conference & Exhibition April 26th & 27th, London

SCADA & SMART GRID CYBER SECURITY SUMMIT 2012

Featuring a two-day Conference, Exhibition, the 'SCADA & Smart Grid Cyber Security Awards 2012' and Networking Gala Dinner

Developments, Strategies and Best Practice in SCADA and Smart Grid Cyber Security

Featuring a two-day Conference with over 25 top level speakers

Discover the latest technologies and solutions for cyber security in the Technology Exhibition

Taking place on the evening of Day One the 'SCADA & Smart Grid Cyber Security Awards 2012' - rewarding achievements and initiatives from utility companies and solution providers

Network with your industry peers and make vital new contacts at the Networking Gala Dinner

Assess the nature of the latest threats being faced by energy companies and the impact of these upon your organisation

Discover why Utility Cyber Security has been reaching a state of near chaos and the latest strategies from utilities to gain the upper-hand against hackers

Understand the importance of industrial control system (ICS) security and assess the latest solutions on offer.

Discuss the most promising cyber security technologies in the marketplace

Assess the trends to watch in utility cyber security

Discover the best practice from across Europe in protecting SCADA and the Smart Grid from cyber-attack

Benefit from case study presentations from a wide range of international utilities and energy companies

Network with your industry peers in the comfort of a 5 star venue

UNMISSABLE
the only conference and training event of its kind to cover SCADA security

Featuring 3 Interactive Training Workshops!

By popular demand from Utilities this year's event will include a selection of 3 not to be missed training workshops on SCADA and Smart Grid Cyber Security

For further details on how to attend the SCADA & Smart Grid and Cyber Security Summit 2012 as a Delegate, Exhibitor or Sponsor then please visit:

Cyber Styletto

6 a.m., Thursday, Yvonne's Key West Bungalow

Yvonne poured through the maliciously functioning server while Colin slept the night's activities off. It was tough working with a hangover, but nothing she hadn't done before – just a little harder to focus was all. The pounding, claustrophobic sensation left over by her partying made her imagine the server as a cave, with its dark secrets hidden at the bottom of some impossibly obscure passage. Cyber spelunker, she thought, and started to laugh, but laughter was painful this early. A Bloody Mary might not solve the computer mystery, but it would take the edge off her discomfort.

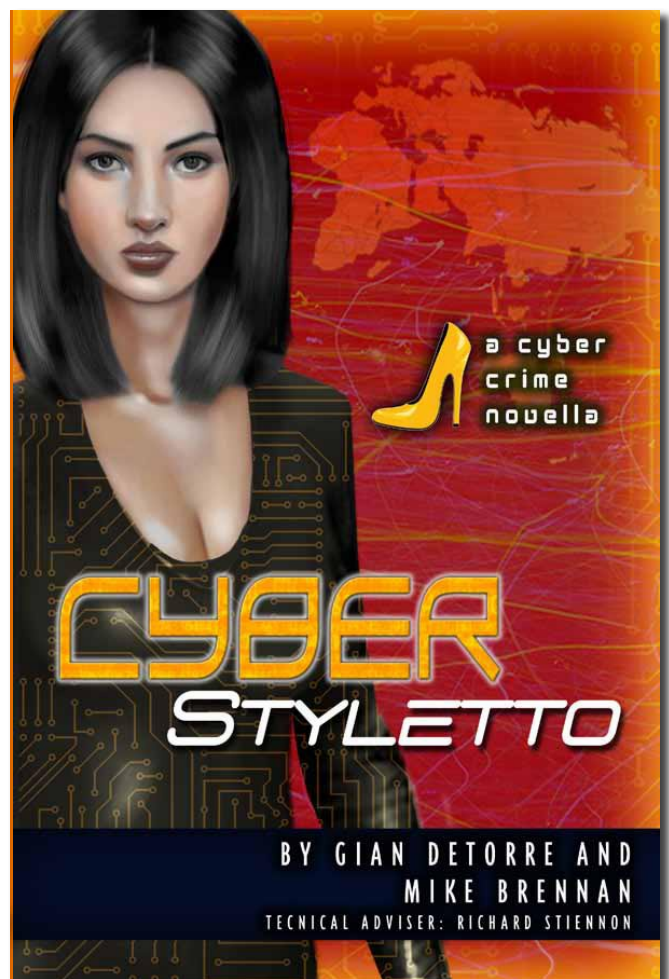
Colin, still naked, met her at the bar. He grabbed her wrist and pulled her towards him, and tried to slip his other hand under her tshirt, but an elbow to his Adam's apple put a stop to that.

"Hey, honey, what's the matter?"

"Sorry, Col. But duty calls. I've got a renegade chip in that server and Stokes is waiting for an answer."

"You sure?" He turned towards her and flexed his pecs. That wasn't where she was looking, however. "You know you want to," he said.

Yes, she did. But the last thing she needed was another suitor, another hanger-on, calling her when she was working, interrupting when she relaxed with someone else. Her life alone was as good as it could be – apart from her responsibilities to CyberCom she made the decisions about where she went and with whom. She made the kind of money most people could only dream of, had the kind of talent and respect others could never even imagine. She wasn't going to let that be tied down in a relationship, no matter how good looking or important the man was. She wouldn't do it for Colin, and she certainly wouldn't do it for Stokes. If only he were here so she could throw an elbow his way.



"Get dressed, Colin. Bring me a Bloody Mary and then you can make us some breakfast. There's eggs and bacon in the fridge. And don't forget the Tabasco."

He did as he was told. At least dating younger men meant they listened occasionally. But that only reminded her of her age. She didn't want to be anyone's mommy, either. Maybe that's why she'd let Lillian go off to live with her family all those years ago – considering how she'd been treated by the relatives

as a child it might not have been the best decision, but then a life spent ignored while mommy glued her gaze to a monitor wouldn't have been good for Lillian either. Yvonne had sent enough money back to Moscow to make sure Lillian received the best education and never wanted for material needs, but she wondered often about her daughter's emotional state. The letters between them were rare, now that Lillian was almost eighteen. God, she was old enough now to date Colin too.

While he rummaged for pans and plates she turned her attention back to the server. How engrossing, this work. In a sense it was liberating too, because it demanded a level of concentration that made life's problems recede into the subconscious. Never mind the dysfunction of her relatives. Never mind the boys and their neverending games to impress her. This was the chess match. Someone had placed this chip on the motherboard, right under the noses of the Network Systems techs, and hid it and its functions well enough that they never suspected until it turned on and ceded control to a remote commander. And the programming! She hooked the board to her system to read the code embedded within. Layers and layers of commands and parameters, perfectly designed to isolate sub stations and controllers, and avoid detection by feeding dummy readings to the CalTrans sensors. Whoever designed this had to know not only the workings of the server, but those of the entire traffic grid and its stations, in order to know how to shut them down. And the chip made it all easy. It acted as a back door through which the attackers could access at any time, as though they had made a skeleton key that unlocked the traffic control program.

Exquisite.

And as she scanned through thousands of lines of code written in redundant languages – even C and C++ – and saw the programmable logic controller rootkit, she knew this was an offspring of Stuxnet, not nearly as involved, but still capable of incredible damage. There was root kit, there the replacement DLLs for the network interfaces. There was the privilege escalation that made it possible to override any attempt by the authorized user to wrest control by changing the root passwords. Since changing the default passwords would disable the server software completely, Network Systems protocols dictated they remain unchanged, giving the worm the opportunity to proliferate exponentially, throughout the system, into any connected device. A single rogue chip might

produce access to dozens of critical functions. In any server that had been compromised, infiltration was made ridiculously easy, and could clear the way for additional breeches and takeovers, like falling dominoes – perhaps even some that the chip's designers hadn't even imagined. Who knew what other control systems might be on the verge of failure?

Stuxnet had needed a team of the best engineers to program it – a team at a government level – and not just any government. Even this new variant required a level of understanding and capability at the top of the programming food chain. Could another country be backing this effort to infiltrate critical systems? Stuxnet had helped delay Iran's effort to achieve nuclear power and weapons. Now maybe someone was trying to turn the tables and damage America's infrastructure, and as she and hundreds of experts knew, many of the country's various infrastructure components were already in danger of failing even without the push from Stuxnet. This kind of worm, with some additional engineering, could potentially give control of vital systems to a hostile force. Instead of mere failure, these systems could be made to cause damage – both physical and economic. With crucial systems in someone else's control, the nation could be essentially powerless to protect itself.

She texted Stokes and told him she needed to speak with Rita Sanchez and whoever headed logistics at Network Systems. A few seconds later, he was signaling on the Scan-U, and his 3-D image appeared in front of the monitor.

"You're up early, Rohan."

"Frankly, I've been up for hours waiting for you to tell me something that I can relay to the White House. What have you got?"

"This." She held up the evil chip. "Somebody got a little crazy with the soldering gun."

His electronic head turned to look at it. "Yvonne, did I ever mention you have a gift for understatement?"

"Whoever's behind this plopped this chip right into the middle of the motherboard. And it's so small and so well programmed even I might not have noticed its presence. So of course it had no trouble getting past the Network Systems team."

"I'd better get Rita on the line with us."

“Hmmm,” Yvonne said. “Already on a first-name basis?”

Something going on there?”

“Don’t make jokes. You know I don’t go for the super bitch type. I like a woman who likes to have fun. Someone with a sense of adventure.”

“Too bad you’re not very adventurous yourself.”

“What? What do you mean?”

It was just a line, but she had touched a nerve.

“Oh come on, Rohan. You sit at a desk all day and push paper and buttons. Once in a while you run off to a meeting. That’s not exactly my idea of adventurous.”

“Is that why you broke it off with me?”

“Sure, sure, darling. The last time you did field work was when you were with the CIA and arrested me.”

“That was a pretty exciting time.”

“And I was impressed. I’d been able to keep my activities a secret from the government and the mob. You were the only one who figured me out.”

“Yvonne, I’m still trying to figure you out.” Stokes paused, then added, “You know, I could get back out there anytime I want.”

“Oh yes, Rohan. You just tell the Assistant for Homeland Security and Counterterrorism his top man is taking some time off to chase the bad guys. I’m sure he’ll understand.”

Colin called to her as he brought two steaming plates from the kitchen. “I have your order, madame. Would you care for another Bloody Mary?”

Stokes’s holograph pivoted left, as though by doing so it could see into the kitchen. “Who is that?”

“Oh, just a house guest.”

“But he’s not wearing a shirt.”

Yvonne laughed. “You’re lucky he’s wearing pants. Colin, say hello to the nice man.”

Colin stopped and smiled. “Hello, Rohan.” He waved into the screen, and then went back to the bar. She imagined Colin’s projection appearing in Stokes’s office, shaking a slightly transparent hand in his face.

“He doesn’t get to call me Rohan,” Stokes said.

“I’ll remind him next time he’s here.”

Stokes puffed his cheeks as though he’d conceded this round of banter. “All right, Ms. Tran, what else can you tell me about this chip?”

“Not much yet. That’s why I need the Network Systems people. They may be able to help at last.”

“Did you ping the chip’s C and C yet?”

“I’ll do that while you’re getting Ms. Sanchez out of bed...oops! I mean, while you’re getting her on the Scan-U.”

“Goodbye, Yvonne. I’ll signal you in a few minutes.”

She put the Scan-U to sleep and brought her breakfast back into the kitchen to join Colin. She found him instead in the living room, hooked up to the huge flat screen Stokes had sent her when they were having their affair. He’d accessed an online community of Armageddon Squadron.

“I’m kicking some serious butt,” he said. “Commando style.” “Didn’t you get enough of that when you flew drones?”

“There’s never enough when you can smell the kill.”

If the search for the origin of the chip meant physically tracking down the source, his flying skills, both real and virtual, might be an asset. But he was so young; in many ways still a boy. Maybe she would boot him back to the northern Keys instead.

Yvonne watched as he manipulated a computer generated aircraft while shoveling his scrambled eggs. He had the coordination of an athlete, able to accomplish difficult maneuvers without conscious thought, and she could see how he’d been able to fly the Air Force’s drones past enemy defenses and over moon-like terrain to drop ordnance on suspected terrorists. He loved it, practically breathed it. But like her, he found out quickly he could make more money in the private sector. His game really was a game –

online flight simulator tournaments, which had made him financially independent. She wasn't unhappy working for the government – and there was even some time to pick up occasional work from cash-rich corporate clients, not to mention the odd gig siphoning tens of thousands from illegal offshore casinos or other scam artists who managed to stay out of the authorities' reach (and at which Stokes looked the other way). It made a nice balance, even if it wasn't as lucrative as a pure black hat existence. At times it made her feel patriotic, that she was returning the favor America had given her by letting her stay and go to MIT. Maybe he would someday see it the same way, too.

She set to tracing the IP address the chip was programmed to send to. But as expected the trail ran cold short of the final destination. She was able to reach back to Hong Kong, but no further. It reached the headquarters of Cathay Computer Works, but from there she could not traceroute past the firewall doing the network address translation for their private network.

A half hour later Stokes signaled that he had the NetSys executives on the Scan-U. Yvonne logged in to see his image scanning her bungalow, as if looking for Colin again. Two more holographs began to resolve, and Yvonne called up her Cyber Styletto image. Rita Sanchez and another man appeared.

"Ms. Sanchez. Good to see you again."

"Good to not see you again," Sanchez said.

"Ah, yes. Sorry about that. But you know the regulations regarding my identity." She studied the face of the man who materialized along with Sanchez. "And Liang Runnan," Yvonne said.

"Who is that? How do you know me?" he said.

"I know of you. CEO at Pebble Computer in Beijing. Put the company on the map. Then you disappeared in 1997. Rumor has it certain Chinese government officials wanted kickbacks for introductions to foreign investors and you couldn't pay."

"I assure you my departure from that company was a personal decision. I only wished to spend more time with my family."

"It didn't make sense, considering the system of extortion over there. Everybody knows to put something

away for when the government comes calling. Unless it's already been extorted, so to speak."

"What are you insinuating?" Liang asked. "Who is this, please?" Stokes cut in. "Our agent knows better than to spread unfounded gossip, doesn't she?"

"So, now you're working for Network Systems." Yvonne continued as if Stokes hadn't said a word.

"I am Senior Vice President and Director of Logistics and Manufacturing for the Pacific Rim Region," Liang said.

"Wow," Yvonne said. "That must be a big business card."

"What can we do for you, Miss...?" Liang said.

"Actually, it's what I can do for you. The source of your infiltration was a microchip, planted on the motherboard of your compromised server. The chip is controlled by an entity somewhere in China, but I haven't been able to pinpoint it yet. The signal stops in Hong Kong, just a stone's throw from Guangzhou."

"That is where the server is manufactured," Liang said. "A microchip? Are you sure?"

"All I have is the serial number. Can you tell me who made it?" She read the number to him, and Liang searched his database, but he reported no returns. "I will launch a full investigation," he said.

"There isn't time for that," Yvonne said. "Stokes, I'm going to have to track this down the old fashioned way. I'm going to put together a team and go to Hong Kong. The only way we can take it further is to go to the data center at Cathay." She smiled over the possibility of doing the field work herself, getting out of the dark computer labs she haunted and out of reach of Stokes and the other government stiff who'd cramped her style since the arrest and subsequent arrangement.

"Finally," Sanchez said. "Some progress."

"Of course there will be a few expenses not covered in the original agreement," Yvonne said.

"Expenses? With what we're paying you? Out of the question."

“Nothing major,” Yvonne went on. “First-class airfare, luxury hotel accommodations, expense account...”

“She’s only kidding,” Stokes said. “And if there are any extra costs, the government will cover them.” “Oh, Rohan. Shame on you,” Yvonne said. “Adding to the national debt. What would the president say?”

“Just keep the expenses to a minimum,” he said.

When Sanchez and Runnan signed off, Yvonne shed her avatar to finish up with Stokes. “I will need a team, Rohan. You know I’m still a suspect for hacking the W88 caper in China, so I need some people I trust to help me. It’ll be easier to remain undercover if I have the right people to work with.”

“You’re really willing to go back there, Yvonne? If they catch you they might put you on trial anyway, with or without proof.”

“Well, you can’t blame them. If I hadn’t caught them with their pants down over those nuclear warhead designs, they might have invaded Taiwan. I can see why they’d be a little upset.”

“Speaking of having someone’s pants down, is that person I saw at your place earlier still hanging around?”

“You mean Colin? I’ll be bringing him along, if that’s what you want to know.”

Colin called from the living room, “Road trip!”

Yes, he was a boy at heart. “Separate bedrooms, I promise,” she said.

“Bring me the receipts or I won’t cover the hotel.”

“Rohan, darling. Jealous to the end. Listen, I’m going to call an old friend, Buck Ryan. I want him and his crew in on this.”

“Ryan Repo? If he’s still kicking I think it’s a good idea. Get some experience involved.”

“Still kicking? Buck will never retire,” she said. “He has too much fun screwing with deadbeats. You ever see the face of a corporate exec when his airplane is repossessed?”

You can buy the paperback version of *Cyber Styletto*, which includes black and white sketches of the characters, by clicking on LuLu.Com

“Can’t say that I have. Buck still headquartered in San Francisco?”

“As soon as I confirm with him I’ll fly there to plan how we’re going to proceed. I’ll call you as soon as we have it together.” Yvonne’s next call was to Ryan’s cell.

“Perfect timing,” Buck said. “I’ve got a triple seven in Shenzhen that I’m snatching on the twenty-fifth. Provincial official got over his head with an Aussie bank, and we’re going to give him a reverse Christmas present.”

“Don’t forget to leave a lump of coal in his stocking,” Yvonne said.

“Ha! You know I’ve missed that sense of humor of yours. It’ll be great to work with you again. What have you got cooking this time?”

“Buck, you’re going to need to call in your muscle for this job,” she said. “We’re talking cyber war.” ♦

END Chapter Four

By Mike Brennan and Richard Stiennon

Porto, Portugal

18 - 21 April, 2012

WEBIST 2012

***8th International Conference on
Web Information Systems and Technologies***

Regular Paper Submission: November 24, 2011

<http://www.webist.org>

By subscribing to this magazine you will win a 50% discount voucher over the registration fee to WEBIST or CLOSER.

(Offer limited to 3 vouchers per conference)

CLOSER 2012

2nd International Conference on
Cloud Computing and Services Science

18 - 21 April, 2012

Porto, Portugal

<http://closer.scitevents.org>

Regular Paper Submission: November 23, 2011