

SECURITY ENHANCE THROUGH WEB SERVICES

Alexandru Florea ^{*1}
Daniela Alexandra CRIȘAN ²

ABSTRACT:

The Internet is the place where a great part of our economy is being built. Here vendors, customers and partners meet together, interact and collaborate. But, even the Internet has created this unique link, the applications used by companies are built around a variety of programming language, platforms and operating systems that cannot communicate to each other. Moreover, every minute we spend on the Internet we face the problem of data security. Web Services represents the next step in the attempt of a programmer to enhance the security of the data and one of the best solutions a company chooses in order to interconnect its web applications without paying attention to platform compatibility or what programming language is used.

Keywords: Web application, Security, Internet, Web Services

This paper has been financially supported by scientific research within the project entitled “PRACTICAL SCHOOL: Innovation in Higher Education and Success on the Labour Market”, project identified as POSDRU/156/1.2/G/132920. The project is co-financed by the European Social Fund through the Sectorial Operational Programme for Human Resources Development 2007-2013. Investing in people!

1. Introduction

A web application is any type of software which is meant to be interpreted by web browsers and it is usually being programmed using a range of browser-supported languages such as JavaScript, HTML, CSS and other. This type of application is based on client-server communication which implies that a client application requests certain information from the server, and the latter sends a response after analysing the request [1].

Amongst the main benefits of using web applications it is worth mentioning:

- They are installed on a powerful computer (server) and it doesn't depend on the client's hardware resources;
- They can be accessed from any location that is connected to the world wide web;

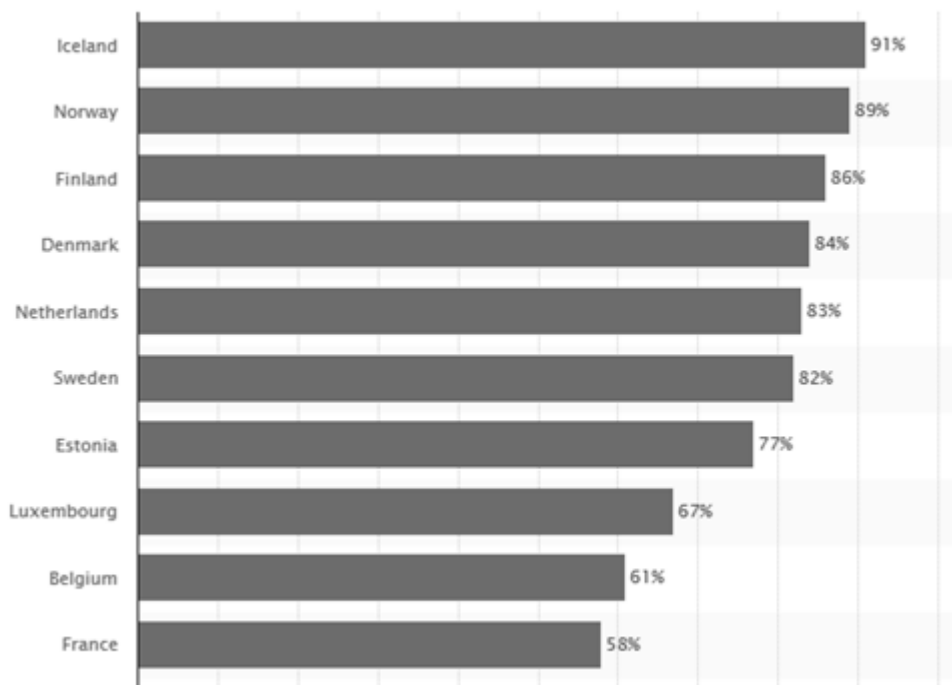
¹ *Corresponding author. School of Computer Science for Business Management, Romanian-American University, 1B, Expozitiei Blvd., district 1, code 012101, Bucharest, Romania. e-mail: alexandru.florea@scenefz.net

² (Coordinator) Associate Professor, PhD, School of Computer Science for Business Management, Romanian-American University, 1B, Expozitiei Blvd., district 1, code 012101, Bucharest, Romania. e-mail: crisan.daniela.alexandra@profesor.rau.ro

- Since they run in a web browser, there needn't be compatibility between the operating systems used by the client and the server;
- They can be run in almost every web browser regardless of its version. However, there are some applications that require a specific web browser or version;

Over the years, the Internet network has been visited by more and more people, regardless of their social status as it represents the cheapest and fastest way of communication and data transfer. The Internet is also an excellent place for online shopping stores, this being highlighted by the fact that in 2006 various companies had over 1 billion online users [2]. Such a big amount of data should be taken over, processed, stored and last but not least provided to users all over the world in a way that is both as fast and accessible as it could be. This is the main reason applications have become indispensable to any viral business.

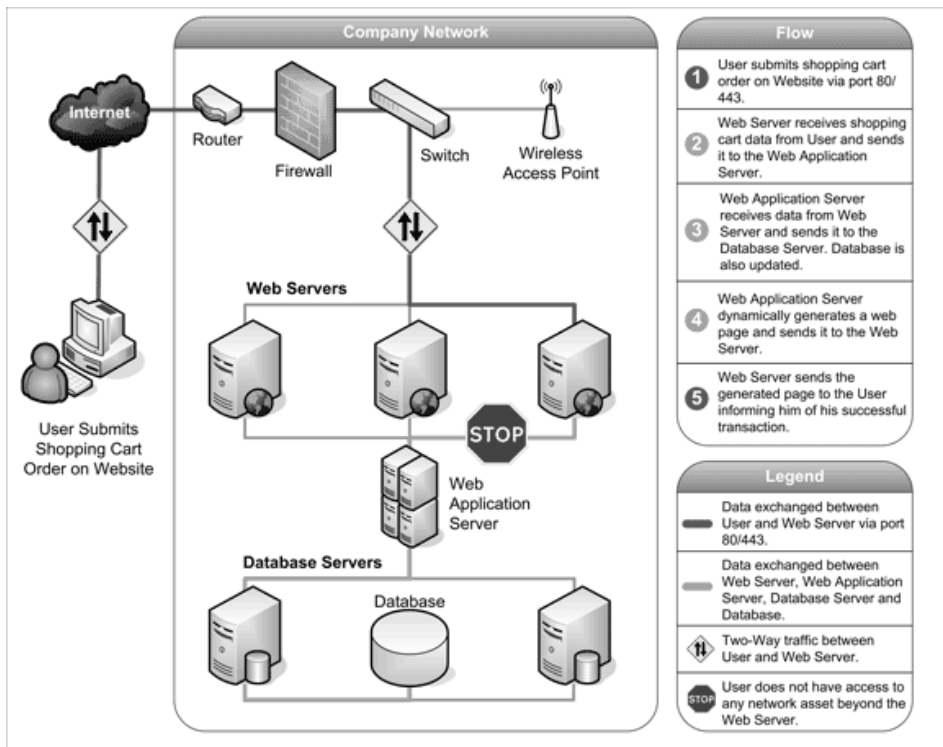
The following chart represents the top 10 European countries to have the most web banking applications users. This study has been carried on 16-74 year-old Internet users by statista.com researchers in 2014.



(source: [3])

Chart 1: Online banking penetration in selected European markets in 2014

- A. Web applications work diagram
- B. The next image points out the way a client sends a request to the server by using a web browser. The web server accesses the database server in order to execute the task. Next, the web application prints to the user through the web browser the information received from the server.
- C.



D.

(source:[16])

Chart 2: Web applications work diagram

E. Security issues regarding web applications

Although web applications are extremely beneficent, they have a high number of issues regarding data security, mostly because of an improper way of programming. This kind of vulnerabilities, such as XSS(Cross-Site Scripting), SQL injection allow hackers to directly access sensitive and personal information stored in the database.

According to The Common Weakness Enumeration (CWE), a community-developed dictionary that catalogues software weaknesses, and SANS Institute, the most common programming errors regarding web applications are (first five from the Top 25 List of the Most Dangerous Software Errors):

Table 1: The most significant errors regarding web applications

Rank	Score	Name
1	93.8	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
2	83.3	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
3	79.0	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
4	77.7	Improper Neutralization of Input During Web Page Generation('Cross-site Scripting')
[5]	76.9	Missing Authentication for Critical Function

(source:[5])

It is being estimated [6] that almost every year there are cybernetic attacks worth \$3.8 million and the cost of data security insurance rises up to \$3.5 million within multinational companies. On average a web application goes through approximately twelve security breaches. This has led to 73% of multinational companies servers being attacked over the past two years.

Another disadvantage of web applications is the fact that they depend on the server's operating system and the programming language used. This means that a PHP application cannot communicate with a C# one. This fact cannot be neglected since all programming languages are uniformly distributed, as it is shown in the graphic below:

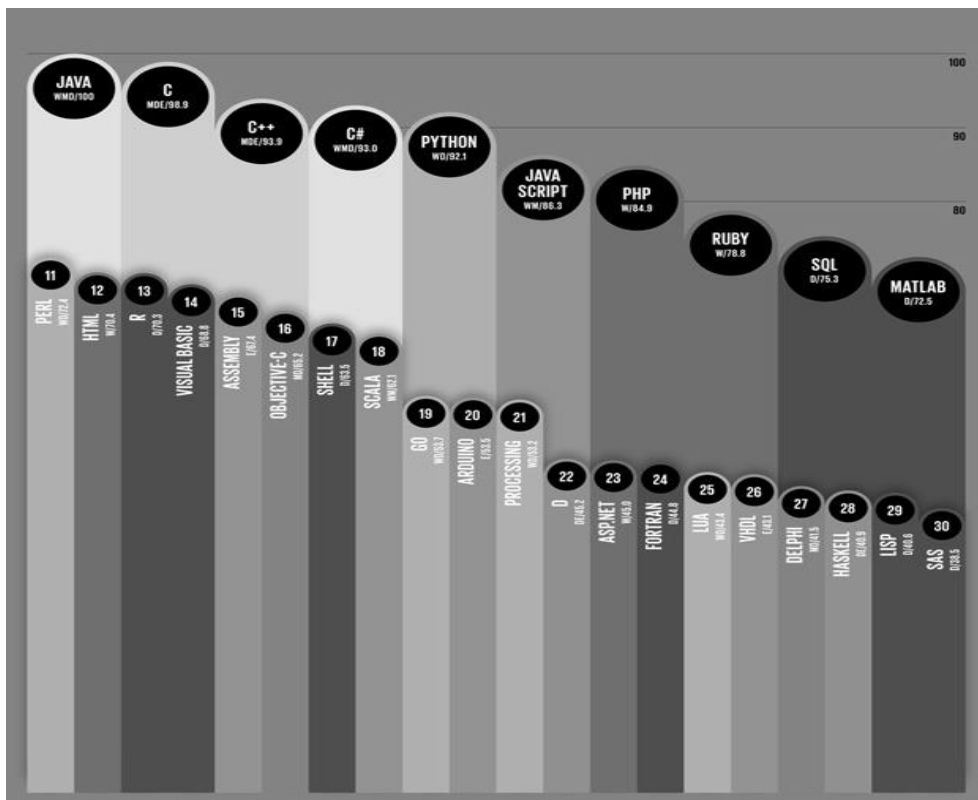


Chart 3: Programming language spread in 2014 (source:[7])

where: W – Web, M – Mobile, D – Desktop and Enterprise, E – Embedded.
Scores are normalized so that the top-ranked language's score is set to 100.

2. Web Services

Web Services are application components that communicate using open protocols such as SOAP (Simple Object Access Protocol), which was designed in order to be accessed by any other application regardless of the programming language used or its operating system.

2.1. Advantages of using Web Services

Five significant advantages of using Web Services have been identified:

- They are platform independent: meaning that the C# stocks application that run on Windows Server will easily communicate with the supplier's PHP commands application running on UNIX;
- They are reusable: there is no need writing custom service for every client, one service or portion of service can be used by several clients;
- They are cheap: reusing the Web Services leads to saving time. Moreover money can be saved by not investing in software and hardware to cover all platforms;
- They enhance security by intermediating client-server communication;
- They are increasingly used in newer sites. The graphic below shows how many Drupal users started to implement Web Services in their web applications since 2012.



(source:[9])

Chart 4: Drupal's usage statistics for Web Services between July 2012 and April 2015

2.2. Web Services structure

Web Services are built by using four technologies: XML, SOAP, WSDL and UDDI.

a) XML (eXtensible Mark-up Language)

XML is a W3C (World Wide Web Consortium) specification that defines a meta-language for describing data. It is a mark-up language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable [8].

In XML applications, data is described by surrounding it with customizable, text-based tags that give information about the data itself as well as its hierarchical structure.

b) SOAP (Simple Object Access Protocol)

SOAP is a XML based communication protocol which ensures the connectivity between applications. It is platform and language independent, which means the applications can run on different operating systems, with different programming languages and technologies. SOAP can operate on transfer protocols like HTTP, SMTP, TCP, UDP and JMS, HTTP being one of the most used protocols.

c) WSDL (Web Service Language Description)

WSDL is a describing language for Web Services. It is a document written in XML that specifies the location of the service and the methods given to use in the application. Because it provides an XML format, it can be interpreted by any programming language.

d) UDDI (Universal Description, Discovery and Integration)

UDDI is a directory service that provides Web Service for any company and possibility to publish their own services and to define how other application will interact with them. It is described in WSDL and communicates using SOAP protocol.

2.3. How do Web Services Work?

Usually, a normal web application works relatively easy. The user sends a request through an interface to the server, and then the server sends back a response with the required data. Because almost all applications use a database to store the information, users must know some login credentials (username and password) to gain access to data. This means that in his request the client application must include the username and password of the database.

But the Internet is not safe: before it gets to the server, the request travels through thousands of nodes, routers and switches where it could be caught and information stolen. It is well known the fact that such sensitive information are sent through a secure protocol like HTTPS (Hypertext Transfer Protocol Secure), but GOOGLE recently announced that they have discovered a security breach of this protocol, so HTTPS is 100% secure no more. The chart above shows how a normal application works.

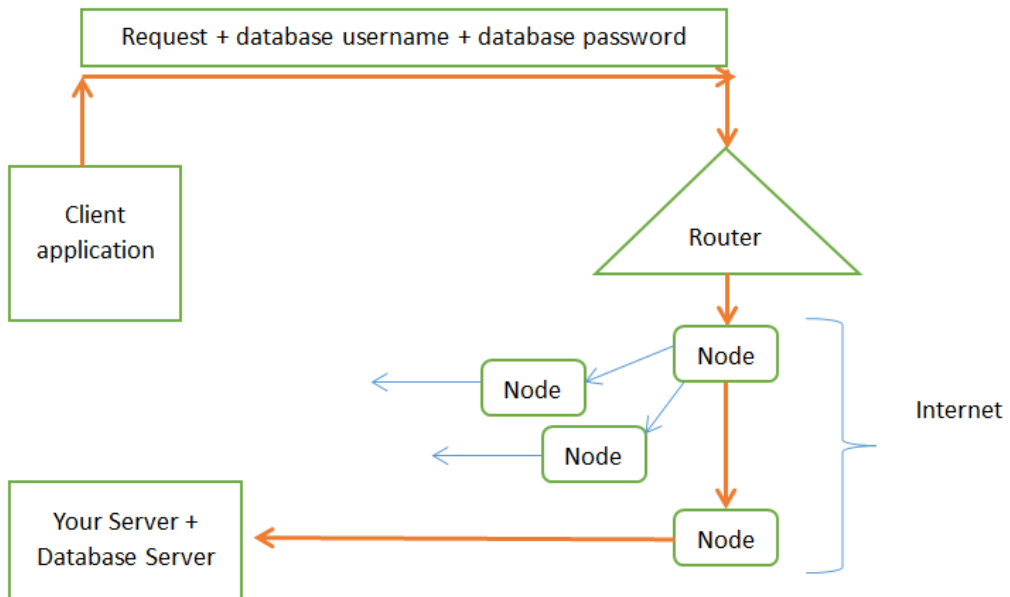


Chart 5: A normal web application

If there's no 100% safe method (so far), the only way would be not sending those sensitive information through internet. Here's the point where Web Services could help. They act like an intermediate between client application and the server. Instead of sending credentials through Internet, access to the database can be given only to Web Services. In order to retrieve data, the client will send a request to Web Services by calling one of the defined methods that has been offered to him. The Web Services will analyse the request and if it corresponds with the methods that have been defined, they will login to the database server, ask for information, get them and then send a response back to the client application. In this way, sending databases sensitive credentials through Internet will be avoided, because Web Services are on the same physical server or in same area network. In order to improve security, a firewall to deny database access to any other client may be installed, letting only Web Services request pass through. The chart above shows how an application using Web Services works.

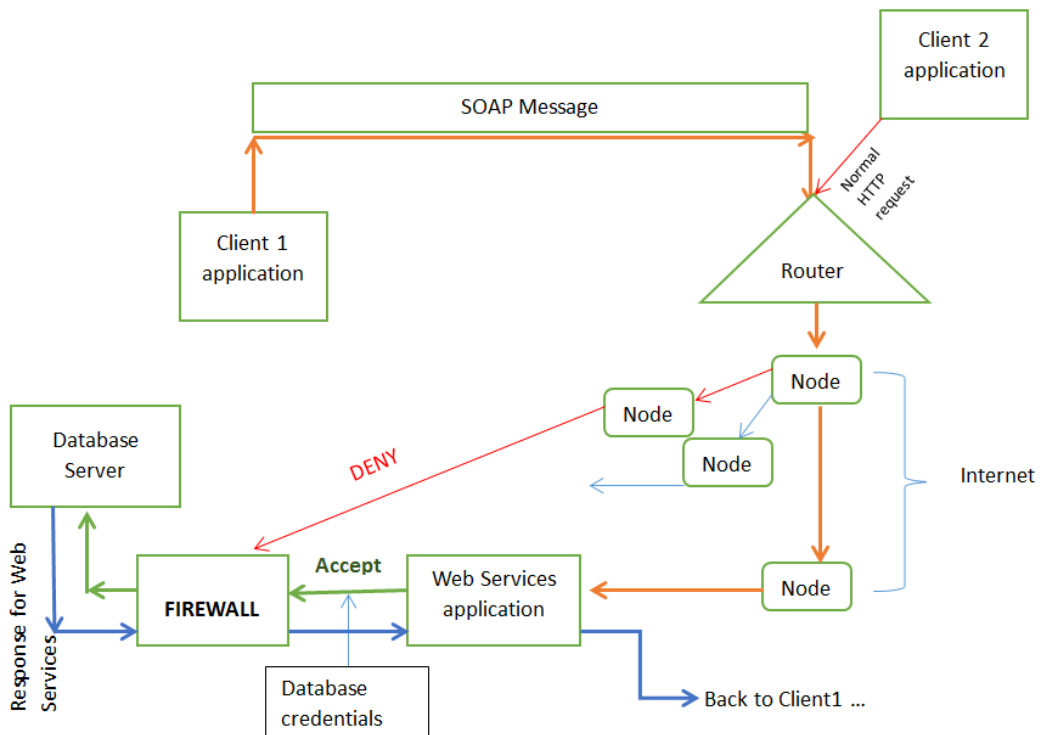


Chart 6: How a Web Services application works.

2.4. Disadvantages of Web Services

There are some minor disadvantages that Web Services have:

- They use a plain text protocol to communicate which is fairly verbose and the request is quite large. Usually it is not a performance issue unless a low-speed internet connection is used or the network is very busy;
- Like HTTP and HTTPS, Web Services are stateless, it means that once the exchange with the client no one will keep a track of each other or have knowledge of each other's state;
- Because of their generality, their purpose is to be offered to a large area of clients, but there will always be a specific customer who needs a specific feature that nobody else needs;
- Any time the Web Services methods or even their parameters are changed it will affect all customers and their applications will crash;
- They are not easily understood. It takes a lot of hard study for a programmer in order to develop proper Web Services.

3. Conclusions

The most important thing a programmer should focus when he develops a new application is the security of the data. Because of their advantages the most used applications are the web based one. We cannot ask our clients to change their way to interact with Internet services by using other kind of applications that are quiet secured so it is programmer's duty to ensure the security of data. Web Services are the next step in Internet Technology that offers to customers safe, scalable and reusable methods in order to develop their online business.

The evolution of Web Services and the fact that they have become more and more used in the past years demonstrates their reliability and a raised level of data protection. Even they use a verbose way to communicate that makes them quiet slow, thanks to the evolution of our world wide web we are able to pass through this issue and ensure that our online services are the best choice of our customers.

REFERENCES

- [1] Daniel Nation – Web Applications
http://webtrends.about.com/od/webapplications/a/web_application.htm.
- [2] Computer Industry Almanac, 2006.
- [3] Online banking penetration in selected European markets in 2014
<http://www.statista.com/statistics/222286/online-banking-penetration-in-leading-european-countries/>.
- [4] How do web applications work? <http://www.acunetix.com/wp-content/uploads/2012/10/web-apps-2.gif>
- [5] Erik Couture, Web Application Injection Vulnerabilities
- [6] mykonossoftware.com statistics, <http://www.mykonossoftware.com/statistics.php>
- [7] IEEE Spectrum Ranking 2014, <http://spectrum.ieee.org/computing/software/top-10-programming-languages>
- [8] Erin Cavanaugh, Web services: Benefits, challenges, and a unique, visual development solution
- [9] Drupal's usage statistics for Web Services since 2012,
<https://www.drupal.org/project/usage/restws>
- [10] Bray, Tim (2004-10-28). "WS-Pagecount". TBray.org
(<http://www.tbray.org/ongoing/When/200x/2004/09/21/WS-Research>)
- [11] The Java EE6 Tutorial Part No: 821–1841–16 January 2013
- [12] Spring in Action, Second Edition, Craig Walls with Ryan Breidenbach, August 2007
- [13] SOA Using Java Web Services, Mark D. Hansen– May 19, 2007
- [14] Java SOA Cookbook, Eben Hewitt, March 2009
- [15] RESTful Web Services, Leonard Richardson, Sam Ruby, May 2005
- [16] Acunetix, <http://www.acunetix.com/wp-content/uploads/2012/10/>