Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium, Volume 21, No. 1, ISSN 1726-9679
ISBN 978-3-901509-73-5, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2010
Make Harmony Between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals of DAAAM International

SECURING THE CLIENT-SERVER COMMUNICATION IN WCF

PALKA, J[an]; PALKA, J[iri] & MOTYL, I[vo]

Abstract: *This paper is aimed to analyse the possibilities of securing the communication inside the Windows Communication Foundation or communication between clients using WCF. WCF offers various bindings, types of security and authentication. Each way was implemented into n-tier test application, tested and its performance was measured. The communication ways used by WCF, binding types varying by used protocols, encoding and specific features were compared.*
Key words: *WCF, authentication, authorization, WAS*

1. INTRODUCTION

The very modern Microsoft technology concerning communication among applications and serving unified programming model for creating applications using services is called WCF (Windows Communication Foundation). The WCF is a SOAP message-based distributed programming platform, and securing messages between clients and services is essential for protecting data. WCF provides a versatile and interoperable platform for the secure exchanging messages based upon both the existing security infrastructure and the recognized security standards for SOAP messages.

In WCF platform it is possible to interoperable with existing technologies such as Windows integrated security, HTTPs, user names and passwords to authenticate users. The WCF brings an addition to existing security infrastructures by using secure SOAP messages. It is possible to use credentials that identify a client or a service, such as user name and password or X.509 certificates and have the interoperable XML-based SOAP profiles. Using these profiles, messages are exchanged securely by taking advantage of open specification like XML digital signatures and XML encryption.

The aim of this study was to determine the methods for securing client-server communication using the windows communication technology.

2. MATERIALS

2.1 Testing environment

The study was built on an experimental system written in C# language, based on the technology .NET 3.5. (Ferracchiati, 2008). This system was developed in the Microsoft Virtual PC system because it gives the possibility to change the hosting operation system if needed and save necessary time for these changes. Especially the changes on the server side are very easily applicable in VPC because it is possible to make various versions of hosting VPC.

2.1 Evaluated system

The evaluated system hosted in the virtual machine consisted of three parts.

- Client – application logic
- Server – service, exposed endpoints
- Core – service code

First part contained the application logic of the client which was connected to the server based application. This service on the server had exposed endpoints used for connecting the clients. It was possible to implement the server application logic into this service but it had some difficulties like necessary to restart this service in the case of changing the code. Because of this, the service code was separated in the third part of the system which was linked as a library in the second part.

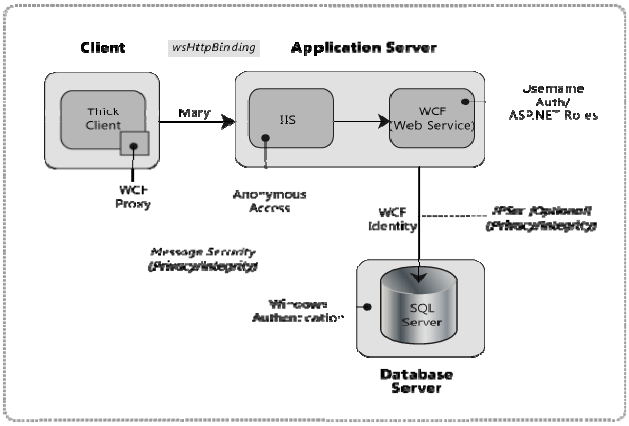

Fig. 1. Testing system structure

Finally it is important to mention that the service part was hosted in the internet information service. This technique was possible because of the WAS feature presence.

2.2 Windows Process Activation Service

Windows Process Activation Service (commonly known as WAS) is built in the existing IIS 6.0 process and hosting models, but brings some advantages as support for other protocols besides HTTP, such as TCP and Named pipes.

By hosting the Windows Communication Foundation (WCF) services in WAS, the application takes advantage of WAS features such as process recycling, rapid failover protection, and the common configuration system, all of which were previously available only to HTTP-based applications.

For testing purposes sample data from Microsoft SQL database were transferred. The sample consisted of 100 table rows with 20 columns with various data types transferred hundred times in a loop. Each measure was repeated ten times to get better comparable results.

3. RESULTS

The client and server communication depends on amount of transferred data, type of chosen binding and type of securing method. The communication is also affected by quality of network connection and characteristics of used active elements in the network like routers and firewalls. These effects were not tested because the test server and client were on the same machine and were connected locally. In the four experiments the test system was set up to use the net.tcp and https bindings

secured by message and transport security with two different types of authentication.sh the headings up to the top of the same column as its text. In view of the tight page constrains, however, do please make the fullest possible use of the text area.

3.1 Local connection

Connection type	Time(s)
tcp.net - message - username	10.17
tcp.net - transport - windows	8.53
https - message - username	10.04
https - transport – windows	10.22

Tab. 1. Local connection speed test

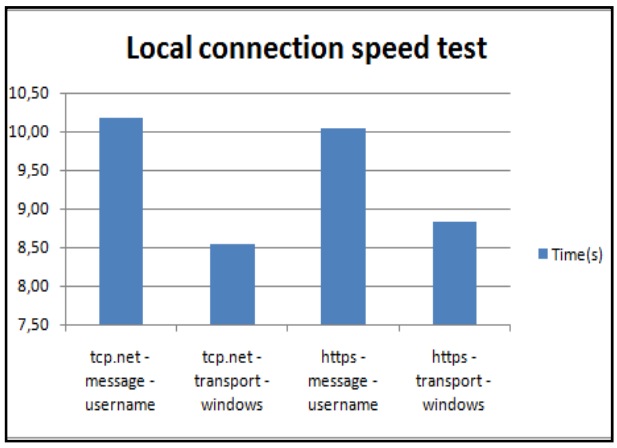

Fig. 2. Local connection speed tests

As can be seen from Table 1 and Fig 2, HTTPS binding was not depending on the type of security and measured times were not markedly different. In the case of tcp.net binding the results were quite different. The average transfer time difference between message and transport types of security was 1.64s that was more than 16% reduction in the case of transport type with windows authentication(Resnick, S., Crane R, 2008).

3.2 Remote connection

Connection type	Time(s)
tcp.net - message - username	19.84
tcp.net - transport - windows	5.42
https - message - username	40.09
https - transport – windows	39.94

Tab. 2. Remote connection speed test

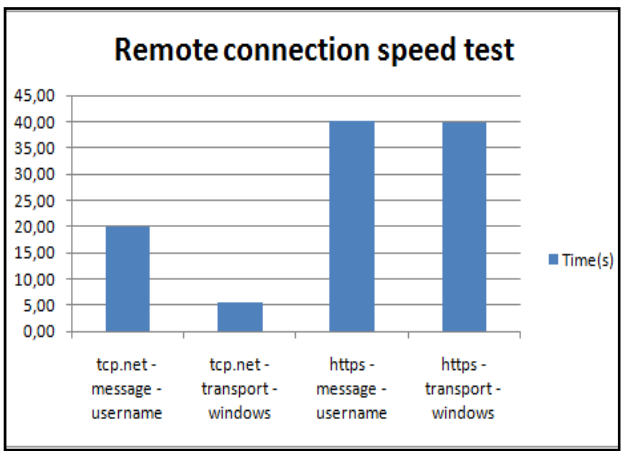

Fig. 3. Remote connection speed tests

The difference between tcp.net and https connections is much more noticeable when the client is connected to the server remotely. In this test the test client application was connected to the server through the internet with several firewalls and routers on the communication channel. In this case the tcp.net connection speeds stayed at the same level. It is necessary to mention that the tcp.net binding with transport type of security has measured faster times than the same communication in the local case, which is caused by more powerful configuration of the server than the client test machine.

It is much more markable speed difference between the tcp.net and https protocols in the remote alternative because in this case the communication is influented by channel quality and of course the remote connection has a slower transfer rate. These facts in the combination with large necessary amount of data needed for message type communication causes into the slower communication times. In this case the difference between these two bindings was about 34.67s which is incredible 639.7% grow. As we can see from the measurement the HTTPs binding has a large speed handicap and should be used in a small data transferring communication only or for purposes where the net.tcp communication could not be used. The tcp.net has a problems in networks with security limitations like firewall port restrictions (McMurtry, 2007).

4. CONCLUSION

This study was carried out to determine the possibilities of securing the communication between client and server connected through the windows communication foundation. Each way of possible connection which the windows communication foundation offers was implemented into n-tier test application. The test data sample was transferred using this test system and the performance was measured. The bindings using different transfer protocols and encoding were also compared. This measurement was made for two types of connection. Firstly, it was supposed that the client is communicating with its server part locally. The windows communication foundation is used as a communication layer in this case only. In the second test were measured times necessary for transfer data through the WCF with connected client over the internet.

This research was done as an entry point for the deep research of various types of communication in different network environments. This planned research should lead to the creation of application design patterns which should be used as a descriptive model showing the correct and most effective way of programming the client-server communicating applications.

5. ACKNOWLEDGEMENTS

This research was supported by the Internal grant agency (IGA/31/FAI/10/D)

6. REFERENCES

Ferracchiati, C. (2008). *LINQ for Visual C#*, First Press, 978-1590598269.

Liu M. (2008) *WCF Multi-tier Services Development with* LINQ, Packt Publishing, 978-1847196620

McMurtry, C.; Mercuri, M., Watling, N. & Winkler, M. (2007). *Windows Communication Foundation - Unleashed.* Sams Publishing, 0672330245

McMurtry, C., Mercuri, M. & Watling, N. (2006) *Microsoft Windows Communication Foundation: Hands-On*, SAMS Publishing, 0-672-32877-1

Resnick, S., Crane R., Bowen S. (2008): *Essential Windows Communication Foundation .Net Framework 3.5.* Addison Wesley, 978-0321440068