Paweł Rajba
pawel@ii.uni.wroc.pl
http://kursy24.eu/

Application Security
Web Services Security

Agenda

- Web Services introduction
- XML Web Services Security
- WS-I Basic Profile, WS-Security
- WCF
  - WCF security basics
  - HTTP Bindings Difference
  - Security Modes
  - Client credential types
  - Protection Level
  - Identities
  - Service Security Context
  - Authorization
  - Debugging
- RESTful services

Web Services

- We consider 2 types of web services
  - XML Web Services
    - Security is mature, different scenarios covered
    - Very widespread, especially in enterprise
  - RESTful services
    - Only several approaches possible
    - Very popular nowadays, especially in the mobile world

XML Web Services Security

- Many specifications support security
  - WS-Security
  - XML Signature, XML Encryption
  - XML Key Management (XKMS)
  - WS-SecureConversation
  - WS-SecurityPolicy
  - WS-Trust, WS-Federation
  - WS-Federation Passive and Active Requestor Profile
  - Web Services Security Kerberos Binding
  - Web Single Sign-On Interoperability Profile
  - Web Single Sign-On Metadata Exchange Protocol
  - Security Assertion Markup Language (SAML)
  - XACML
- During this presentation we focus on WS-Security

More: http://en.wikipedia.org/wiki/List_of_web_service_specifications

WS-I Basic Profile

- WS-I stands for Web Services Interoperability
  - Home page: http://www.ws-i.org/
- A subset of WS specifications, that is supported by most of vendors
- The list of framework compliant with BP one can find here: http://en.wikipedia.org/wiki/WS-I_Basic_Profile
- Basic Profile 2.0
  - http://ws-i.org/profiles/BasicProfile-2.0-2010-11-09.html
- Basic Security Profile 1.0
  - http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html

WS-Security

- Extension to SOAP which applies security
- Provides mechanisms for ensuring
  - Integrity (signing)
  - Confidentiality (encrypting)
- Most commonly allowed methods
  - Username/Password
  - X.509
  - SAML
  - Kerberos
- Specifications including different profiles are available on the OASIS pages:
  - https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss

WCF

- Windows Communication Foundation
- SOAP message-based distributed programming platform
- Supports WS-Security and other specifications
- Endpoints represent access point to service
  They consists
  - Address
  - Binding
  - Contract
- Two main bindings based on HTTP
  - BasicHttpBinding (related to WS-I)
  - WsHttpBinding (related to WS-Security)

WCF Security Basics

- The following security concepts are applied in Windows Communication Foundation:
  - Authentication
  - Authorization
  - Integrity
  - Confidentiality

HTTP Bindings Difference

Criteria	BasicHttpBinding	WsHttpBinding
Security support	This supports the old ASMX style, i.e., WS-BasicProfile 1.1.	This exposes web services using WS-* specifications.
Compatibility	This is aimed for clients who do not have .NET 3.0 installed and it supports wider ranges of clients. Many of the clients like Windows 2000 still do not run .NET 3.0. So an older version of .NET can consume this service.	As it is built using WS-* specifications, it does not support wider ranges of clients and it cannot be consumed by older .NET versions less than 3 version.
SOAP version	SOAP 1.1	SOAP 1.2 and WS-Addressing specification.
Reliable messaging	Not supported. In other words, if a client fires two or three calls you really do not know if they will return back in the same order.	Supported as it supports WS-* specifications.
Default security options	By default, there is no security provided for messages when the client calls happen. In other words, data is sent as plain text.	As `WsHttBinding` supports WS-*, it has WS-Security enabled by default. So the data is not sent in plain text.
Security options	• None • Windows – default authentication • Basic • Certificate	• None • Transport • Message • Transport with message credentials

Source: http://www.codeproject.com/Articles/36396/Difference-between-BasicHttpBinding-and-WsHttpBind

Security Modes

- The 5 modes are available
  - None
    - No security
  - Transport
    - Security on transport level both for authentication and message protection
  - Message
    - Security on message level both for authentication and message protection
  - Both
    - Security on message and transport levels both for authentication and message protection (supported only by MSMQ)
  - TransportWithMessageCredentials
    - Credentials passed with the message
    - Message protection and server authentication provided by the transport layer
  - TransportCredentialsOnly
    - Credentials passed on transport layer
    - No message protection

Client credential types

- We consider two levels
  - On transport level
    - None
    - Basic
    - Windows
    - Certificate
  - On message level
    - None
    - Windows
    - UserName
    - Certificate
    - IssuedToken
      - Notice: BasicHttpBinding supports only UserName and Certificate because of specification Basic Security Profile per WS-I limitation

Protection Level

- Options
  - None, Sign, and EncryptAndSign
- It is related to message level
- Examples in code

```
[ServiceContract(ProtectionLevel = ProtectionLevel.EncryptAndSign)]
public interface IDummyService
{
    [OperationContract]
    string GetData(int value);
}

[ServiceContract]
public interface IDummyService
{
    [OperationContract(ProtectionLevel=ProtectionLevel.Sign)]
    string GetData(int value);
}
```

Identities

- Process Identity
  - ASP.NET account used to run WCF process under e.g. IIS
- Security Principal
  - Caller's identity
- ServiceSecurityContext
  - Provides security runtime context, e.g. caller's id or authorization context

Service Security Context

- Security information in runtime
  - AuthorizationContext
    - Mainly access to claims
  - PrimaryIdentity
    - Identity from claims, especially in case non-integrated security model
  - WindowsIdentity
    - Identity provided as a result of integrated security process

Authorization

- Authentication
  - Element <serviceCredentials> in the web.config
  - Related to client credential types described previously
- Authorization
  - Element <serviceAuthorization> in the web.config
  - Main types
    - UseWindowsGroups
    - UseAspNetRoles
      - Requires Membership and Role Providers
    - Custom
      - Requires implementing custom service authorization manager

Debugging

- We can use Fiddler in the following way
  - Let's assume that our service is run under port 14666
  - Run Fiddler and configure it as a reverse proxy following the first section of instructions from
    - http://docs.telerik.com/fiddler/Configure-Fiddler/Tasks/UseFiddlerAsReverseProxy
      - Please note that adding **ReverseProxyForPort** key inside HKEYCURRENTUSER\SOFTWARE\Microsoft\Fiddler2 is required
  - Reconfigure console client to connect with port 8888 instead of port 14666
- Disclaimer: works only if server is not verified

DEMO

- UserPass*
- Membership*
  - Review certificates on web server
- Windows*

RESTful services

- REST stands for REpresentationalStateTransfer
- Basic concepts
  - Resources represented by URIs
    - Address
    - Resource
    - Parameters
  - HTTP methods as operations/contract
  - HTTP headers used for
    - Content negotation
    - Status codes
    - Security, etag, customheaders
  - HTTP body
- Common data formats: JSON and XML
- Examples
  - POST http://example.com/srv/invoice/123
  - GET http://example.com/srv/invoice/123

RESTful services

- RESTful services can be implemented by
  - WCF
  - WebAPI
- Security model depends on chosen technology
  - However, it is important to remember, that RESTful services should **stateless**

DEMO

- BasicREST
  - Review authorization filter
  - Client as a Fiddler

References

- WS-Security Specifications
  - https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss
- Fiddler as a reverse proxy
  - http://docs.telerik.com/fiddler/Configure-Fiddler/Tasks/UseFiddlerAsReverseProxy
- OWASP WCF Security Best Practices
  - https://www.owasp.org/index.php/WCF_Security_Best_Practices
- WCF Security Explanations
  - http://visualstudiomagazine.com/Articles/2013/08/01/Security-Considerations-and-Best-Practices-for-WCF-4-Apps.aspx?Page=2
  - http://www.codemag.com/article/0611051
  - http://msdn.microsoft.com/en-us/library/ms977327.aspx
  - http://msdn.microsoft.com/en-us/library/ms977312.aspx
  - http://msdn.microsoft.com/pl-pl/library/ms732362(v=vs.110).aspx
  - http://msdn.microsoft.com/en-us/library/ff648318.aspx
- WCF Bindings
  - http://www.codeproject.com/Articles/36396/Difference-between-BasicHttpBinding-and-WsHttpBind
  - http://stackoverflow.com/questions/2106715/basichttpbinding-vs-wshttpbinding
- WCF Examples
  - http://www.microsoft.com/en-us/download/details.aspx?id=21459