Implementation Details:

Implementation of Needham Schroeder Symmetric Key Protocol using Microsoft .Net Framework (Using SOAP protocol for message exchange between nodes using Microsoft Webserivces).

Tools Used:

The project is implemented using Visual Studio 2008 available for download at Stony Brook IT Website

http://it.stonybrook.edu/software/title/microsoft-visual-studio

For Profiling of the project I used **JetBrains – dotNetTrace** tool and code review is done using **FxCop** tool.

Instructions to run the file:

Steps:

1.) Open ./ AuthenticationService/AuthenticationService.sln in Visual Studio 2008 and execute the project. This will launch the webservices for Authentication Server.
2.) Open ./ Receiver/ Receiver.sln in Visual Studio 2008 and execute the project. This will launch the webservies for Receiver (B).
3.) Finally Open the ./ Sender/ Sender.sln in Visual Studio 2008 and execute the project. A client application will appear on which you can click on the Authenticate Button to authenticate the client by consuming webservices hosted in step 1 and 2.

Scenarios Implemented:

1.) Implemented Needham–Schroeder Symmetric key protocol flawed version with both AES and DES encryption.
2.) Implemented Needham–Schroeder Symmetric key protocol fixed version with both AES and DES encryption.

Modules:

We have five major modules in the implementation of the project.

Authentication Server

Authentication server exposes web services which a client (Sender) will consume to get the authenticate token and session key to start communication with another node (Receiver) in the network.

Receiver

Receiver also exposes web services to communicate with the Sender and to establish session after accepting authentication token send by Sender.

Sender

Sender is the main client application which consumes the web services exposed by Authentication Server and Receiver to establish a session with the Receiver.

For better user experience I have implemented the authentication process and User Interface on different threads so that the user can see the exchange of keys on UI without blocking of main thread .

Helper

Helper project is inherited by other projects (Authentication Server, Receiver and Sender) to provide helper functions like `encryptMessage`, `decryptMessage`, `getNodeValue`, `getXMLForAuthenticationServer` etc.

XML Contracts

XML Contracts contain Contract Classes which define how (defines XML format) Sender will be consuming the services exposed by Authentication Server and Receiver.

Class Diagrams:

Authentication Server

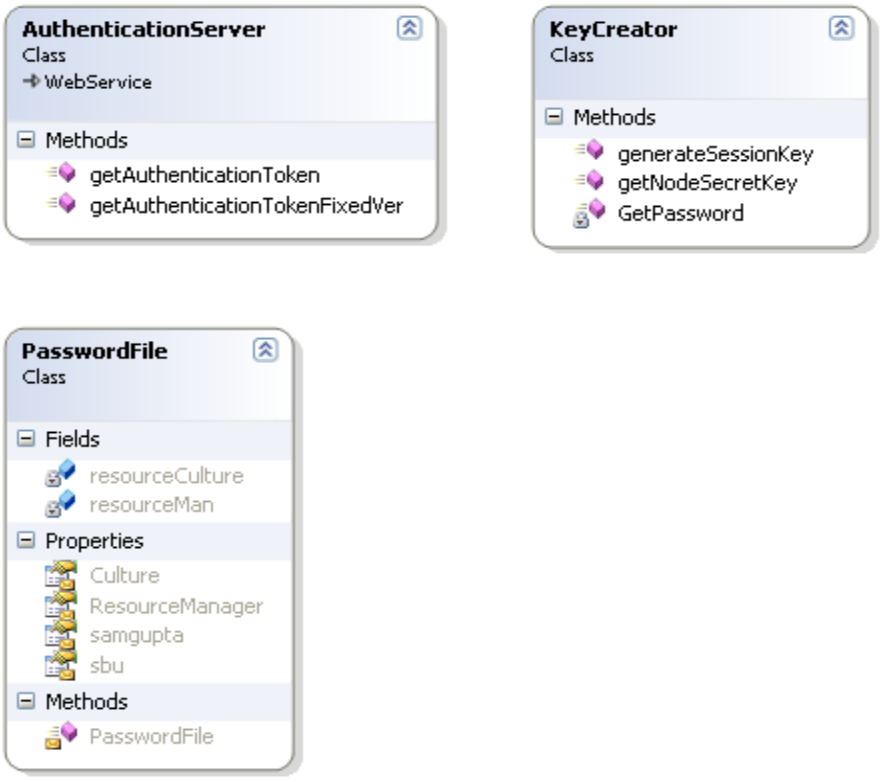
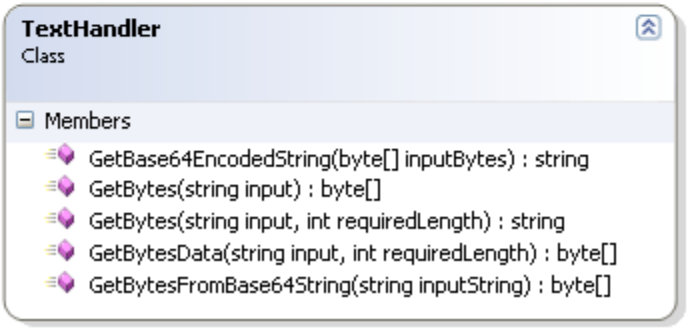


Figure 1

Helper



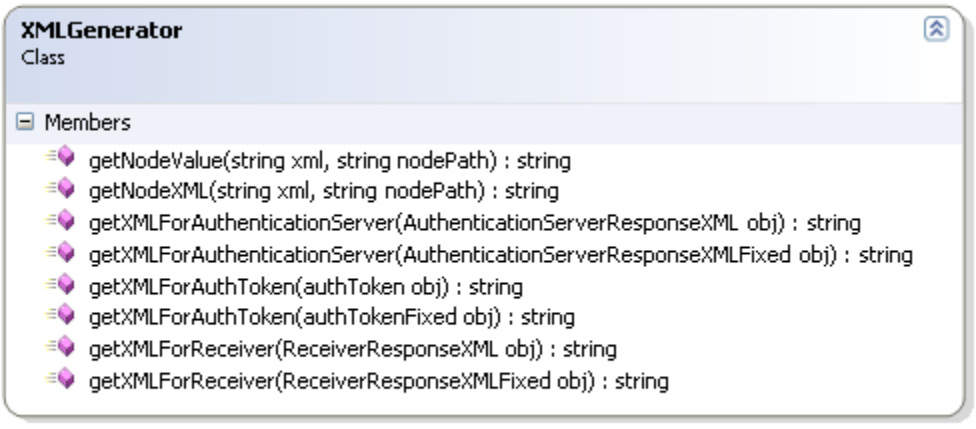


Figure 2

Receiver

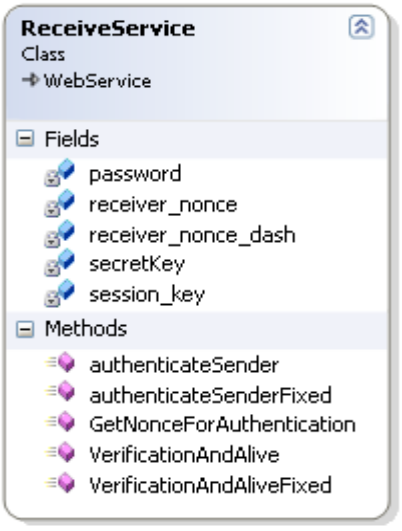
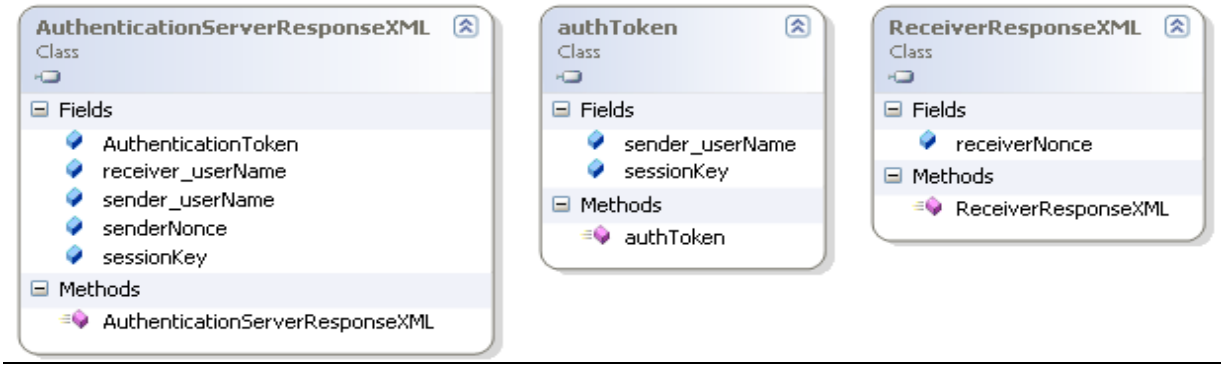


Figure 3

XML Contracts



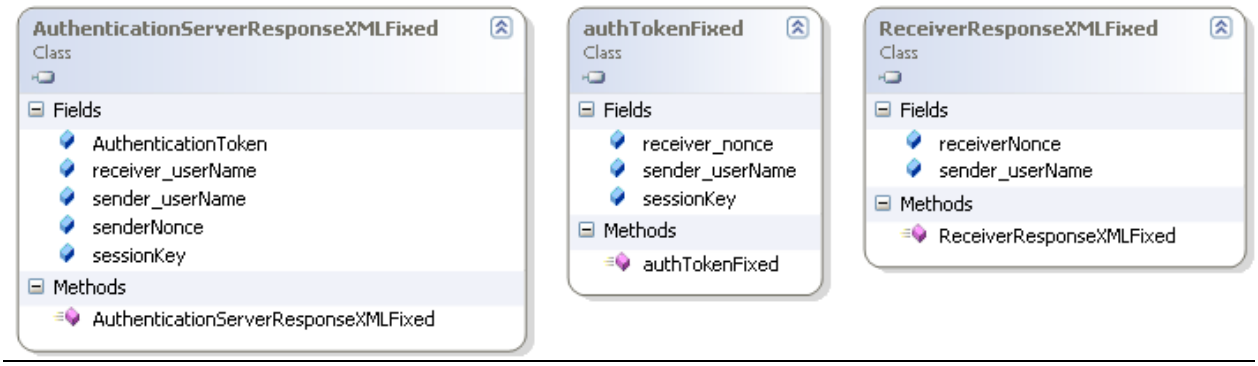
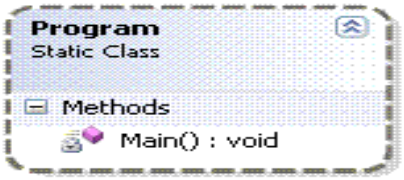


Figure 4

Sender

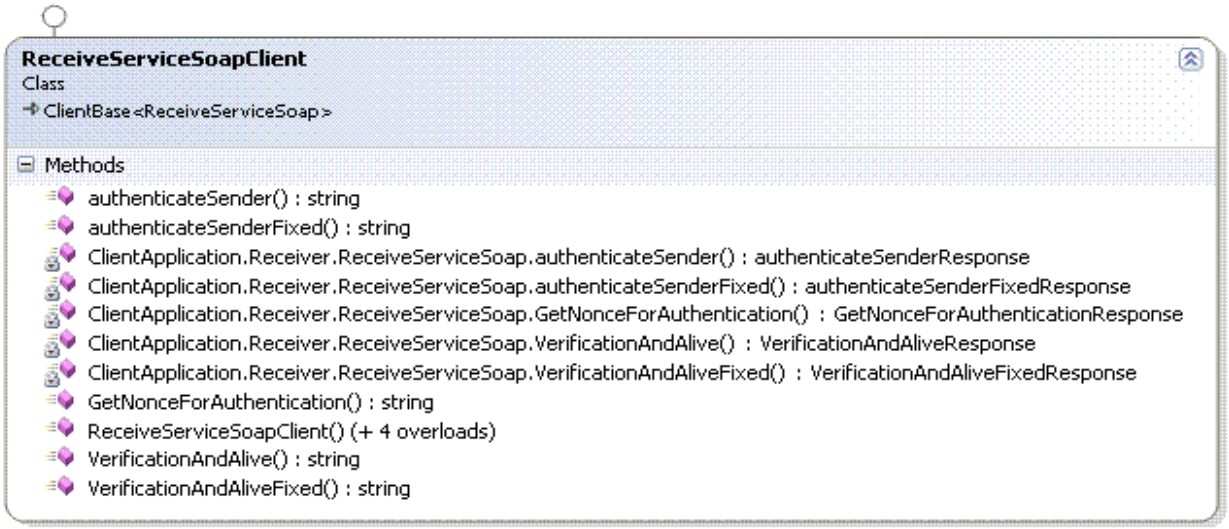

Program
Static Class
Methods
Main() : void

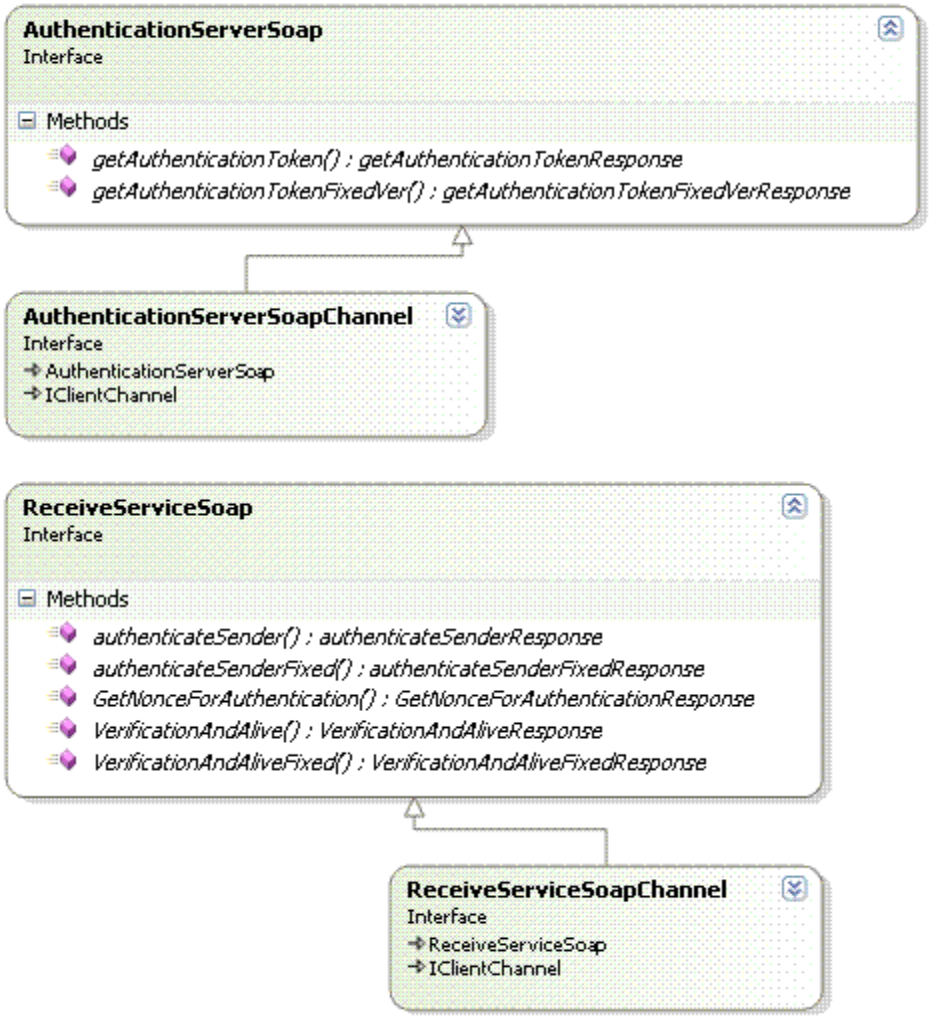

FixedVersionDesign
Class
Methods
ackVerificationAndAlive() : string
GetNonceForAuthentication() : string
requestAuthentication() : string
requestSessionKey() : string


EncryptionType
Enum
AES
DES
OpenSSL


FlawedVersionDesign
Class
Methods
ackVerificationAndAlive() : string
requestAuthentication() : string
requestSessionKey() : string


AuthenticationServerSoapClient
Class
ClientBase<AuthenticationServerSoap>
Methods
AuthenticationServerSoapClient() (+ 4 overloads)
ClientApplication.AuthenticationServer.AuthenticationServerSoap.getAuthenticationToken() : getAuthenticationTokenResponse
ClientApplication.AuthenticationServer.AuthenticationServerSoap.getAuthenticationTokenFixedVer() : getAuthenticationTokenFixedVerResponse
getAuthenticationToken() : string
getAuthenticationTokenFixedVer() : string


ReceiveServiceSoapClient
Class
ClientBase<ReceiveServiceSoap>
Methods
authenticateSender() : string
authenticateSenderFixed() : string
ClientApplication.Receiver.ReceiveServiceSoap.authenticateSender() : authenticateSenderResponse
ClientApplication.Receiver.ReceiveServiceSoap.authenticateSenderFixed() : authenticateSenderFixedResponse
ClientApplication.Receiver.ReceiveServiceSoap.GetNonceForAuthentication() : GetNonceForAuthenticationResponse
ClientApplication.Receiver.ReceiveServiceSoap.VerificationAndAlive() : VerificationAndAliveResponse
ClientApplication.Receiver.ReceiveServiceSoap.VerificationAndAliveFixed() : VerificationAndAliveFixedResponse
GetNonceForAuthentication() : string
ReceiveServiceSoapClient() (+ 4 overloads)
VerificationAndAlive() : string
VerificationAndAliveFixed() : string


AuthenticationServerSoap
Interface
Methods
getAuthenticationToken() : getAuthenticationTokenResponse
getAuthenticationTokenFixedVer() : getAuthenticationTokenFixedVerResponse
AuthenticationServerSoapChannel
Interface
AuthenticationServerSoap
IClientChannel
ReceiveServiceSoap
Interface
Methods
authenticateSender() : authenticateSenderResponse
authenticateSenderFixed() : authenticateSenderFixedResponse
GetNonceForAuthentication() : GetNonceForAuthenticationResponse
VerificationAndAlive() : VerificationAndAliveResponse
VerificationAndAliveFixed() : VerificationAndAliveFixedResponse
ReceiveServiceSoapChannel
Interface
ReceiveServiceSoap
IClientChannel

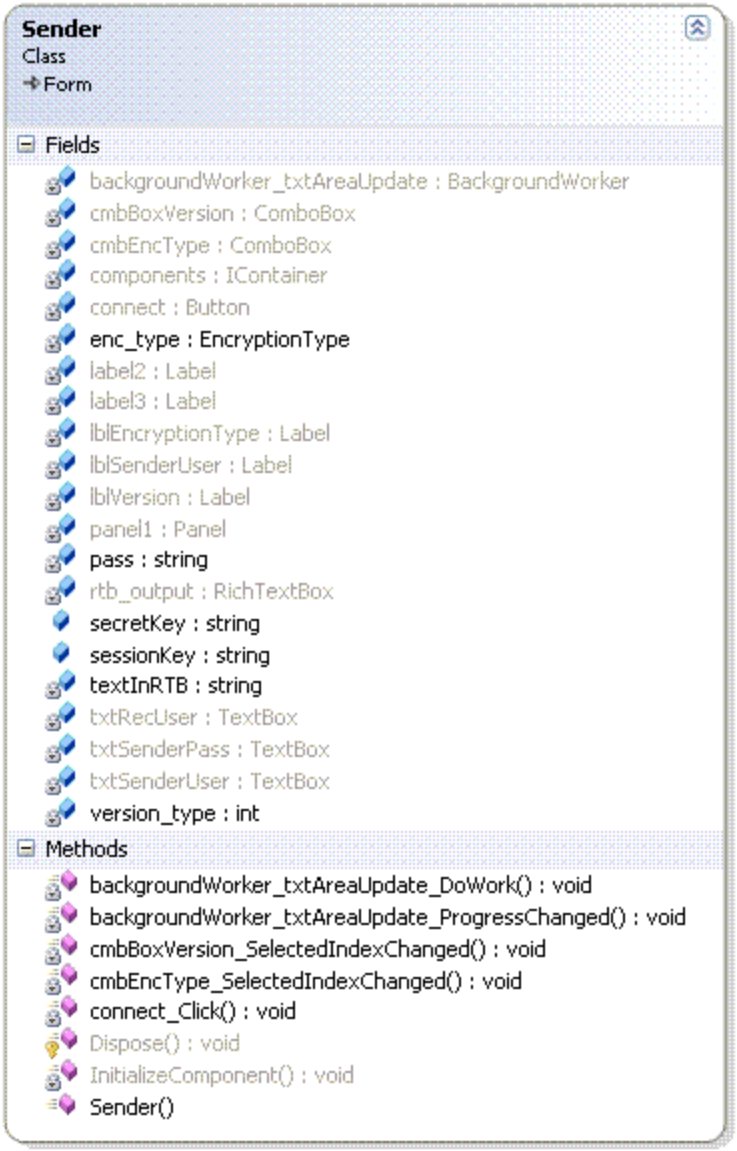

Sender
Class
Form
Fields
backgroundWorker_txtAreaUpdate : BackgroundWorker
cmbBoxVersion : ComboBox
cmbEncType : ComboBox
components : IContainer
connect : Button
enc_type : EncryptionType
label2 : Label
label3 : Label
lblEncryptionType : Label
lblSenderUser : Label
lblVersion : Label
panel1 : Panel
pass : string
rtb_output : RichTextBox
secretKey : string
sessionKey : string
textInRTB : string
txtRecUser : TextBox
txtSenderPass : TextBox
txtSenderUser : TextBox
version_type : int
Methods
backgroundWorker_txtAreaUpdate_DoWork() : void
backgroundWorker_txtAreaUpdate_ProgressChanged() : void
cmbBoxVersion_SelectedIndexChanged() : void
cmbEncType_SelectedIndexChanged() : void
connect_Click() : void
Dispose() : void
InitializeComponent() : void
Sender()

Figure 5

Screen Shot

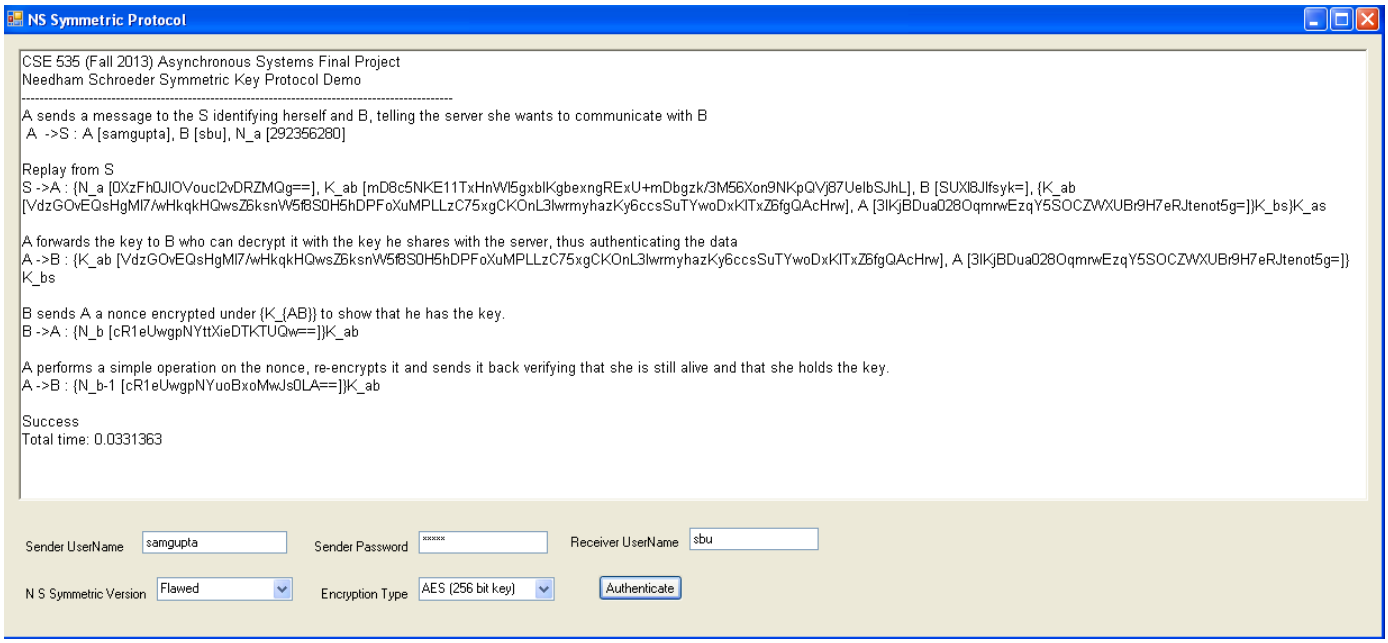


Figure 6

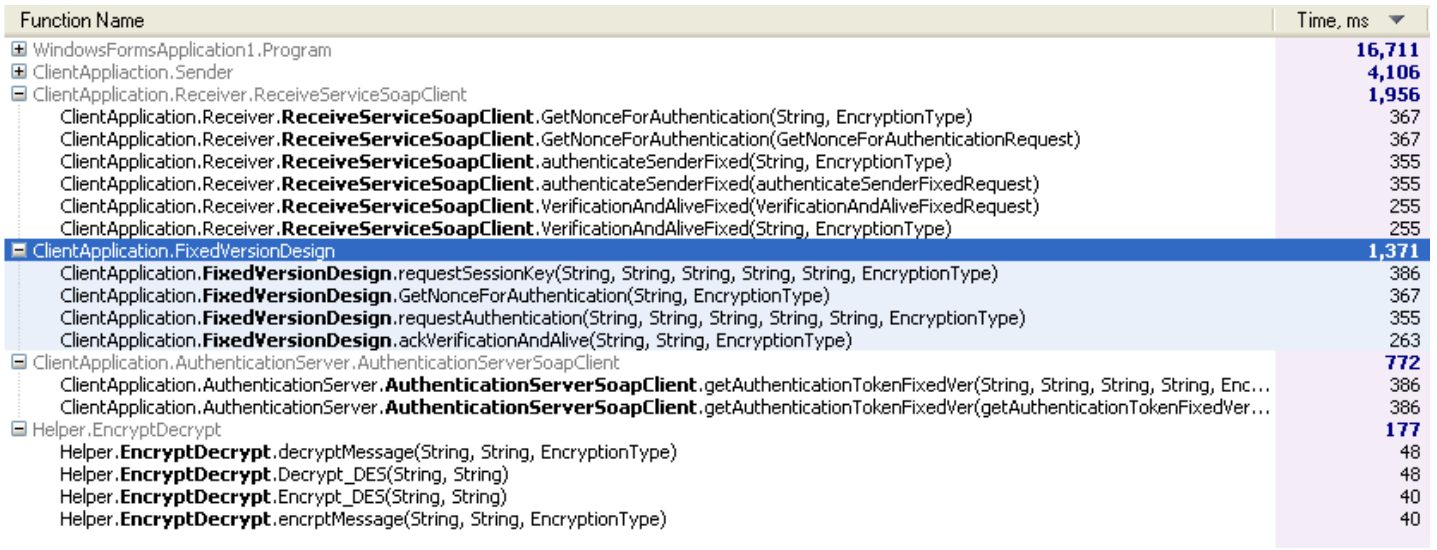
Performance Evaluation

NS Symmetric Key Evaluation using AES Encryption (Flawed Version)

Function Name	Time, ms
WindowsFormsApplication1.Program	19,120
ClientAppliaction.Sender	4,219
ClientApplication.Receiver.ReceiveServiceSoapClient	1,325
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSender(String, EncryptionType)	437
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSender(authenticateSenderRequest)	437
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAlive(VerificationAndAliveRequest)	225
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAlive(String, EncryptionType)	225
ClientAppliaction.FlawedVersionDesign	1,188
ClientAppliaction.**FlawedVersionDesign**.requestSessionKey(String, String, String, String, EncryptionType)	513
ClientAppliaction.**FlawedVersionDesign**.requestAuthentication(String, String, String, String, EncryptionType)	449
ClientAppliaction.**FlawedVersionDesign**.ackVerificationAndAlive(String, String, EncryptionType)	225
ClientApplication.AuthenticationServer.AuthenticationServerSoapClient	1,005
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationToken(String, String, String, EncryptionType)	503
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationToken(getAuthenticationTokenRequest)	503
Helper.XMLGenerator	69
Helper.**XMLGenerator**.getNodeValue(String, String)	57
Helper.**XMLGenerator**.getXMLForAuthToken(authToken)	12
Helper.EncryptDecrypt	37
Helper.**EncryptDecrypt**.decryptMessage(String, String, EncryptionType)	12
Helper.**EncryptDecrypt**.Decrypt_DES(String, String)	12
Helper.**EncryptDecrypt**.Encrypt_DES(String, String)	7
Helper.**EncryptDecrypt**.encrptMessage(String, String, EncryptionType)	7

Figure 7

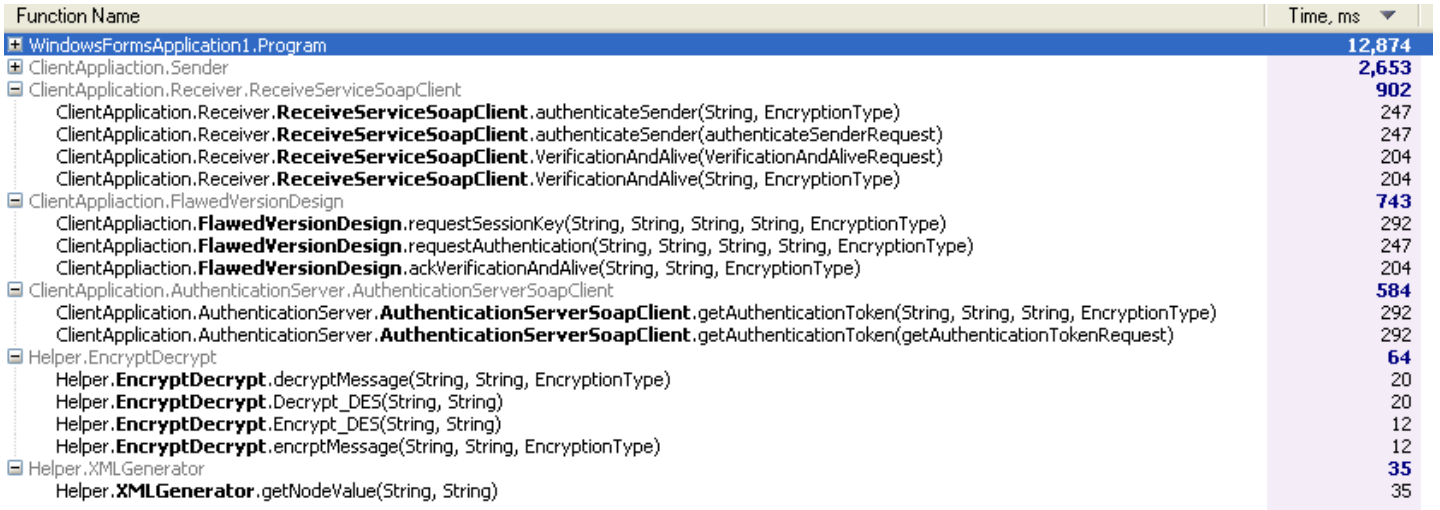
NS Symmetric Key Evaluation using AES Encryption (Fixed Version)



Function Name	Time, ms
WindowsFormsApplication1.Program	16,711
ClientAppliaction.Sender	4,106
ClientApplication.Receiver.ReceiveServiceSoapClient	1,956
ClientApplication.Receiver.**ReceiveServiceSoapClient**.GetNonceForAuthentication(String, EncryptionType)	367
ClientApplication.Receiver.**ReceiveServiceSoapClient**.GetNonceForAuthentication(GetNonceForAuthenticationRequest)	367
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSenderFixed(String, EncryptionType)	355
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSenderFixed(authenticateSenderFixedRequest)	355
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAliveFixed(VerificationAndAliveFixedRequest)	255
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAliveFixed(String, EncryptionType)	255
ClientApplication.FixedVersionDesign	1,371
ClientApplication.**FixedVersionDesign**.requestSessionKey(String, String, String, String, String, EncryptionType)	386
ClientApplication.**FixedVersionDesign**.GetNonceForAuthentication(String, EncryptionType)	367
ClientApplication.**FixedVersionDesign**.requestAuthentication(String, String, String, String, String, EncryptionType)	355
ClientApplication.**FixedVersionDesign**.ackVerificationAndAlive(String, String, EncryptionType)	263
ClientApplication.AuthenticationServer.AuthenticationServerSoapClient	772
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationTokenFixedVer(String, String, String, String, Enc...	386
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationTokenFixedVer(getAuthenticationTokenFixedVer...	386
Helper.EncryptDecrypt	177
Helper.**EncryptDecrypt**.decryptMessage(String, String, EncryptionType)	48
Helper.**EncryptDecrypt**.Decrypt_DES(String, String)	48
Helper.**EncryptDecrypt**.Encrypt_DES(String, String)	40
Helper.**EncryptDecrypt**.encrptMessage(String, String, EncryptionType)	40

Figure 8

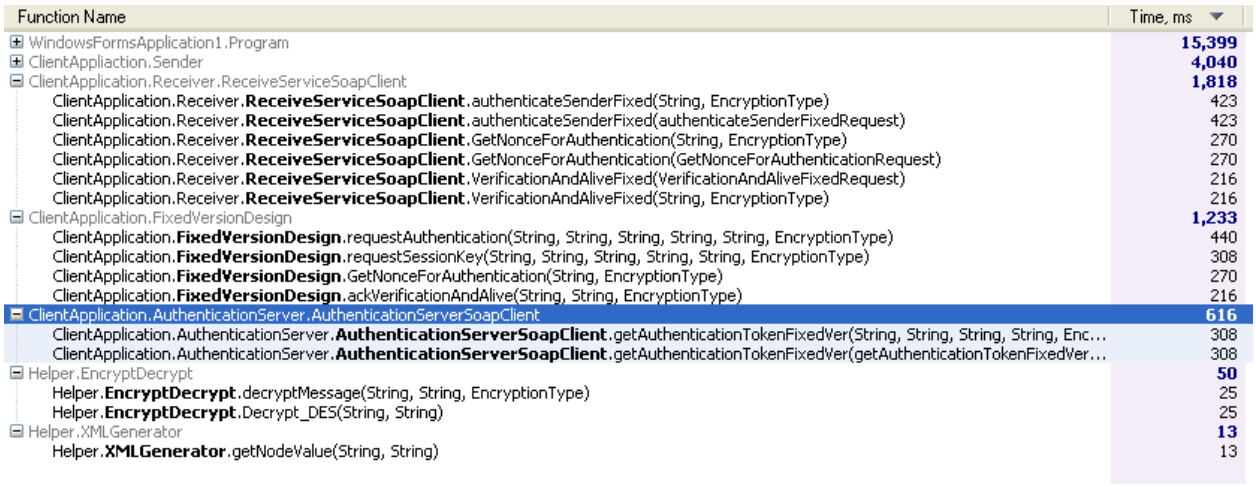
NS Symmetric Key Evaluation using DES Encryption (Flawed Version)



Function Name	Time, ms
WindowsFormsApplication1.Program	12,874
ClientAppliaction.Sender	2,653
ClientApplication.Receiver.ReceiveServiceSoapClient	902
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSender(String, EncryptionType)	247
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSender(authenticateSenderRequest)	247
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAlive(VerificationAndAliveRequest)	204
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAlive(String, EncryptionType)	204
ClientAppliaction.FlawedVersionDesign	743
ClientAppliaction.**FlawedVersionDesign**.requestSessionKey(String, String, String, String, EncryptionType)	292
ClientAppliaction.**FlawedVersionDesign**.requestAuthentication(String, String, String, String, EncryptionType)	247
ClientAppliaction.**FlawedVersionDesign**.ackVerificationAndAlive(String, String, EncryptionType)	204
ClientApplication.AuthenticationServer.AuthenticationServerSoapClient	584
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationToken(String, String, String, EncryptionType)	292
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationToken(getAuthenticationTokenRequest)	292
Helper.EncryptDecrypt	64
Helper.**EncryptDecrypt**.decryptMessage(String, String, EncryptionType)	20
Helper.**EncryptDecrypt**.Decrypt_DES(String, String)	20
Helper.**EncryptDecrypt**.Encrypt_DES(String, String)	12
Helper.**EncryptDecrypt**.encrptMessage(String, String, EncryptionType)	12
Helper.XMLGenerator	35
Helper.**XMLGenerator**.getNodeValue(String, String)	35

Figure 9

NS Symmetric Key Evaluation using DES Encryption (Fixed Version)



Function Name	Time, ms
WindowsFormsApplication1.Program	**15,399**
ClientAppliaction.Sender	**4,040**
ClientApplication.Receiver.ReceiveServiceSoapClient	**1,818**
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSenderFixed(String, EncryptionType)	423
ClientApplication.Receiver.**ReceiveServiceSoapClient**.authenticateSenderFixed(authenticateSenderFixedRequest)	423
ClientApplication.Receiver.**ReceiveServiceSoapClient**.GetNonceForAuthentication(String, EncryptionType)	270
ClientApplication.Receiver.**ReceiveServiceSoapClient**.GetNonceForAuthentication(GetNonceForAuthenticationRequest)	270
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAliveFixed(VerificationAndAliveFixedRequest)	216
ClientApplication.Receiver.**ReceiveServiceSoapClient**.VerificationAndAliveFixed(String, EncryptionType)	216
ClientApplication.FixedVersionDesign	**1,233**
ClientApplication.**FixedVersionDesign**.requestAuthentication(String, String, String, String, String, EncryptionType)	440
ClientApplication.**FixedVersionDesign**.requestSessionKey(String, String, String, String, String, EncryptionType)	308
ClientApplication.**FixedVersionDesign**.GetNonceForAuthentication(String, EncryptionType)	270
ClientApplication.**FixedVersionDesign**.ackVerificationAndAlive(String, String, EncryptionType)	216
ClientApplication.AuthenticationServer.AuthenticationServerSoapClient	**616**
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationTokenFixedVer(String, String, String, String, Enc…	308
ClientApplication.AuthenticationServer.**AuthenticationServerSoapClient**.getAuthenticationTokenFixedVer(getAuthenticationTokenFixedVer…	308
Helper.EncryptDecrypt	**50**
Helper.**EncryptDecrypt**.decryptMessage(String, String, EncryptionType)	25
Helper.**EncryptDecrypt**.Decrypt_DES(String, String)	25
Helper.XMLGenerator	**13**
Helper.**XMLGenerator**.getNodeValue(String, String)	13

Figure 10

Pending Task

I will be hosting the webservices for Authentication server and Receiver on the university server and provide a client application which can be executed with a single click rather than first installing the web services using Visual Studio (As described in the **Instructions to run the file** Step 1 and 2) on your local machine and then launching the client application (Step 3).