using System;

using System.Globalization;

using System.IO;

using EAGetMail; *//add EAGetMail namespace*

static void **Main**(**string**[] args)

{

try

{

*// Create a folder named "inbox" under current directory to save messages*

**string** localInbox = **string**.Format("{0}\\inbox", Directory.GetCurrentDirectory());

*// If the folder is not existed, create it.*

if (!Directory.Exists(localInbox))

{

Directory.CreateDirectory(localInbox);

}

*// Gmail IMAP4 server is "imap.gmail.com"*

MailServer oServer = new MailServer("imap.gmail.com",

"gmailid@gmail.com",

"yourpassword",

ServerProtocol.Imap4);

*// Enable SSL connection.*

oServer.SSLConnection = true;

*// Set 993 SSL port*

oServer.Port = 993;

MailClient oClient = new MailClient("TryIt");

oClient.Connect(oServer);

MailInfo[] infos = oClient.GetMailInfos();

Console.WriteLine("Total {0} email(s)\r\n", infos.Length);

for (**int** i = 0; i < infos.Length; i++)

{

MailInfo info = infos[i];

Console.WriteLine("Index: {0}; Size: {1}; UIDL: {2}",

info.Index, info.Size, info.UIDL);

*// Receive email from IMAP4 server*

Mail oMail = oClient.GetMail(info);

Console.WriteLine("From: {0}", oMail.From.ToString());

Console.WriteLine("Subject: {0}\r\n", oMail.Subject);

*// Generate an unqiue email file name based on date time.*

**string** fileName = \_generateFileName(i + 1);

**string** fullPath = **string**.Format("{0}\\{1}", localInbox, fileName);

*// Save email to local disk*

oMail.SaveAs(fullPath, true);

*// Mark email as deleted from IMAP4 server.*

oClient.Delete(info);

}

*// Quit and expunge emails marked as deleted from IMAP4 server.*

oClient.Quit();

Console.WriteLine("Completed!");

}

catch (Exception ep)

{

Console.WriteLine(ep.Message);

}

}

}

}

**Gmail IMAP OAUTH**

By default, you need to enable [“Allowing less secure apps”](https://support.google.com/accounts/answer/6010255?hl=en) in Gmail/G Suite, then you can retrieve email with user/password IMAP4 authentication.

However Google will disable traditional user authentication in the future, switching to Google OAuth is strongly recommended now.

* [Retrieve Email from Gmail using User Interaction OAuth](https://www.emailarchitect.net/eagetmail/ex/c/20.aspx)
* [Retrieve Email from G Suite using Service Account in Background Service](https://www.emailarchitect.net/eagetmail/ex/c/21.aspx)

**[C# Example - Retrieve Unread/New Email from Gmail Account]**

Because IMAP/EWS/WebDAV support read mail flag, with this feature, we can also retrieve unread/new email only from Gmail like this

using System;

using System.Globalization;

using System.IO;

using EAGetMail; *//add EAGetMail namespace*

namespace receiveemail

{

class **Program**

{

*// Generate an unqiue email file name based on date time*

static **string** **\_generateFileName**(**int** sequence)

{

DateTime currentDateTime = DateTime.Now;

return **string**.Format("{0}-{1:000}-{2:000}.eml",

currentDateTime.ToString("yyyyMMddHHmmss", new CultureInfo("en-US")),

currentDateTime.Millisecond,

sequence);

}

static void **Main**(**string**[] args)

{

try

{

*// Create a folder named "inbox" under current directory*

*// to save the email retrieved.*

**string** localInbox = **string**.Format("{0}\\inbox", Directory.GetCurrentDirectory());

*// If the folder is not existed, create it.*

if (!Directory.Exists(localInbox))

{

Directory.CreateDirectory(localInbox);

}

*// Gmail IMAP4 server is "imap.gmail.com"*

MailServer oServer = new MailServer("imap.gmail.com",

"gmailid@gmail.com",

"yourpassword",

ServerProtocol.Imap4);

*// Enable SSL connection.*

oServer.SSLConnection = true;

*// Set 993 SSL port*

oServer.Port = 993;

MailClient oClient = new MailClient("TryIt");

oClient.Connect(oServer);

*// retrieve unread/new email only*

oClient.GetMailInfosParam.Reset();

oClient.GetMailInfosParam.GetMailInfosOptions = GetMailInfosOptionType.NewOnly;

MailInfo[] infos = oClient.GetMailInfos();

Console.WriteLine("Total {0} unread email(s)\r\n", infos.Length);

for (**int** i = 0; i < infos.Length; i++)

{

MailInfo info = infos[i];

Console.WriteLine("Index: {0}; Size: {1}; UIDL: {2}",

info.Index, info.Size, info.UIDL);

*// Receive email from IMAP4 server*

Mail oMail = oClient.GetMail(info);

Console.WriteLine("From: {0}", oMail.From.ToString());

Console.WriteLine("Subject: {0}\r\n", oMail.Subject);

*// Generate an unqiue email file name based on date time.*

**string** fileName = \_generateFileName(i + 1);

**string** fullPath = **string**.Format("{0}\\{1}", localInbox, fileName);

*// Save email to local disk*

oMail.SaveAs(fullPath, true);

*// mark unread email as read, next time this email won't be retrieved again*

if(!info.Read)

{

oClient.MarkAsRead(info, true);

}

*// if you don't want to leave a copy on server, please use*

*// oClient.Delete(info);*

*// instead of MarkAsRead*

}

*// Quit and expunge emails marked as deleted from IMAP4 server.*

oClient.Quit();

Console.WriteLine("Completed!");

}

catch (Exception ep)

{

Console.WriteLine(ep.Message);

}

}

}

}

## TLS 1.2

TLS is the successor of SSL, more and more Email servers require TLS 1.2 encryption now.

If your operating system is Windows XP/Vista/Windows 7/Windows 2003/2008/2008 R2/2012/2012 R2, and you got connection error with SSL/TLS connection, you need to enable TLS 1.2 protocol in your operating system like this:

[Enable TLS 1.2 on Windows XP/Vista/7/10/Windows 2008/2008 R2/2012](https://www.emailarchitect.net/eagetmail/kb/tls.aspx?cat=0)

**Next Section**

At next section I will introduce how to download emails from Yahoo account.