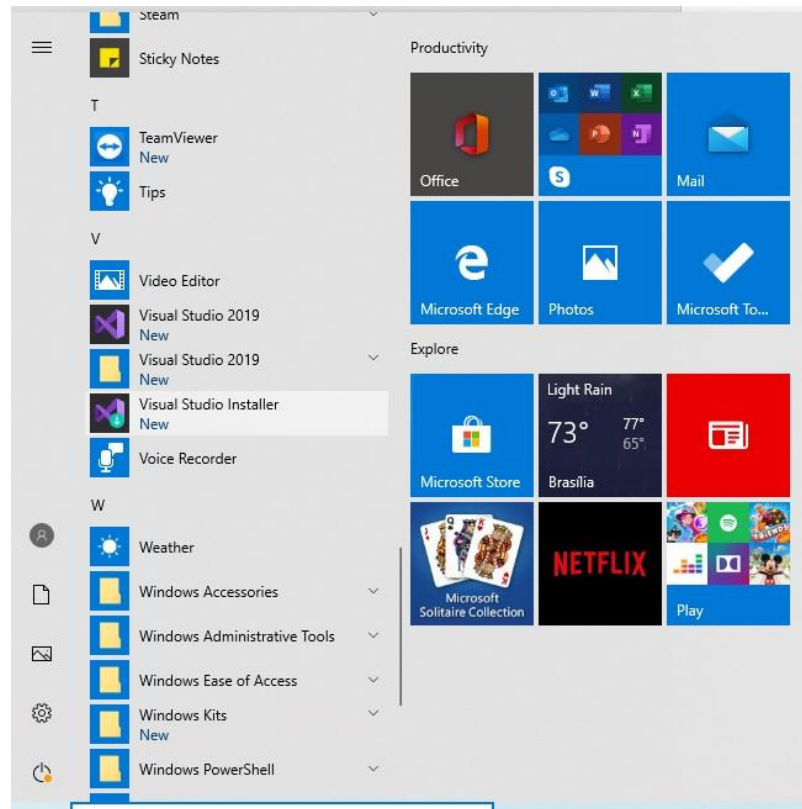


1. Unzip the folder Newman.zip

Unzip the folder to your Desktop, or wherever you like.

2. Make sure that ATL is installed

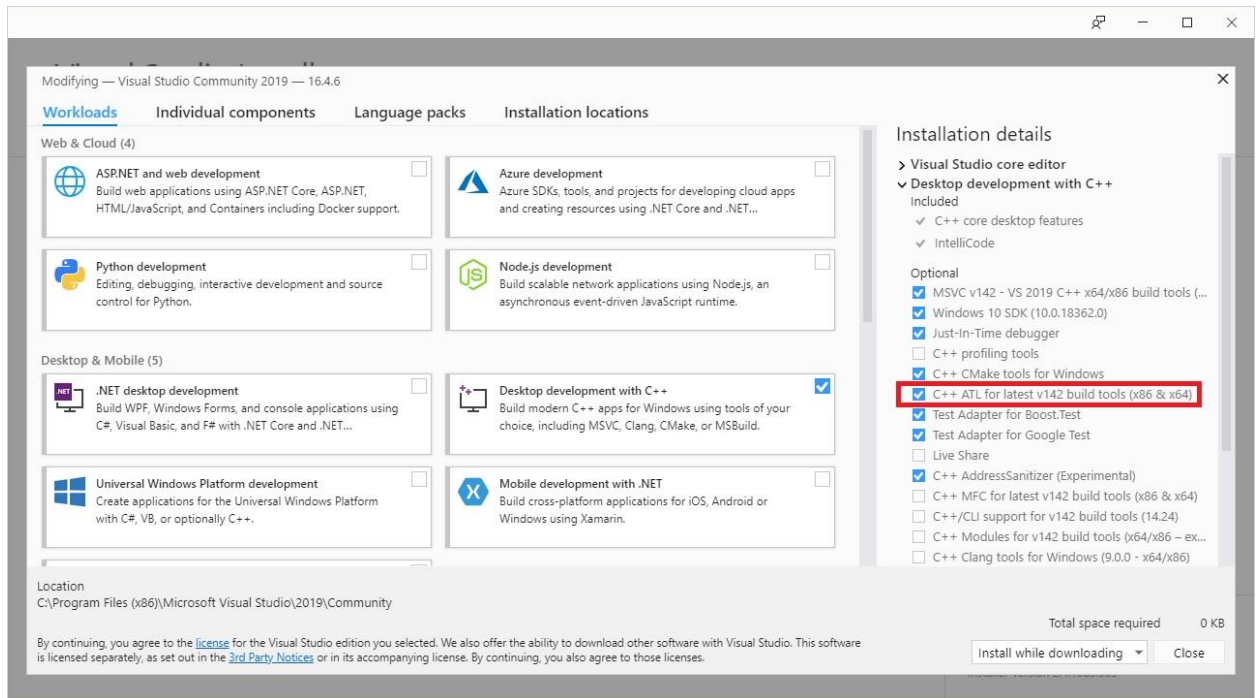
Open up Visual Studio Installer. To do this, go to your **Start**, then look for Visual Studio Installer.



After opening it, click **Modify**.

On the right side, under “**Desktop Development with C++**”, make sure that **C++ ATL for latest v142 build tools** is checked.

If it wasn't checked, then check and install it.



3. Open the command window

Go to Search, or hit **Windows Key+R** and type **cmd** to open the command window.

4. Find the full path to your cmake application

The usual path, if your VS installation is the default, is **C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\CommonExtensions\Microsoft\CMake\CMake\bin\cmake.exe**

If you can't find it there, open a CMakeLists.txt file in Visual Studio 2019 and then go to "CMake Settings" to see where the variable CMAKE_COMMAND points to. That is the path to your cmake.exe

5. Switch back to the command window so we can start compiling

Go to the command window and navigate to the Newman folder where you unzip the downloaded .zip file from my Github. You can navigate to there using the **cd** command.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.657]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Eron>cd Desktop\Newman\Newman

C:\Users\Eron\Desktop\Newman\Newman>dir
Volume in drive C has no label.
Volume Serial Number is 4C9C-1FF0

Directory of C:\Users\Eron\Desktop\Newman\Newman

03/14/2020  12:39 AM    <DIR>          .
03/14/2020  12:39 AM    <DIR>          ..
03/14/2020  12:39 AM    <DIR>          Base64
03/13/2020  11:52 PM             1,003 CMakeLists.txt
03/14/2020  12:39 AM    <DIR>          Hash
03/14/2020  12:39 AM    <DIR>          LibreSSL
03/13/2020  11:00 PM             1,059 LICENSE.txt
03/14/2020  12:39 AM    <DIR>          MessageHeaders
03/13/2020  11:00 PM             2,697 README.md
03/14/2020  12:39 AM    <DIR>          Sasl
03/14/2020  12:39 AM    <DIR>          Smtplib
03/14/2020  12:39 AM    <DIR>          SmtplibAuth
03/14/2020  12:39 AM    <DIR>          src
03/14/2020  12:39 AM    <DIR>          StringExtensions
03/14/2020  12:39 AM    <DIR>          SystemAbstractions
03/14/2020  12:39 AM             432 test.eml
03/14/2020  12:39 AM    <DIR>          TlsDecorator
               4 File(s)              5,191 bytes
              13 Dir(s)  273,041,813,504 bytes free
```

After you are inside Newman folder and with the **dir** command displaying the same as above, create a new directory called **build** with the command: **mkdir build**. Then, enter this directory by typing **cd build**.

Now we will start building with cmake. For that, you will type the full path to your cmake.exe following two dots “..”, like in the image below. You should get the same output. The command, in my case, is:

“C:\Program Files (x86)\Microsoft Visual Studio2019\Community\Common7\IDE\CommonExtensions\Microsoft\CMake\CMa ke\bin\cmake.exe” ..

Note that the double quote in the full path of cmake.exe is needed!!

```
C:\Users\Eron\Desktop\Newman\Newman build
```

```
C:\Users\Eron\Desktop\Newman\Newman\x64 build  
C:\Users\Eron\Desktop\Newman\Newman\build>C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\CommonExtensions\Microsoft\Clake\Clake\bin\cmake.exe --  
Building for: Visual Studio 16 2019  
Selecting Windows SDK version 10.0.18362.0 to target Windows 10.0.18363.  
The C compiler identification is MSVC 19.24.28310.0  
The CXX compiler identification is MSVC 19.24.28310.0  
Check for working C compiler: C:/Program Files (x86)/Microsoft Visual Studio/2019/Community/VC/Tools/MSVC/x64_24.28314/bin/mstools64/x64/cl.exe  
Check for working C++ compiler: C:/Program Files (x86)/Microsoft Visual Studio/2019/Community/VC/Tools/MSVC/x64_24.28314/bin/mstools64/x64/cxxcl.exe -- works  
Detecting C compiler ABI info  
Detecting C compiler ABI info - done  
Detecting C compile features  
Detecting C compile features - done  
Check for working CXX compiler: C:/Program Files (x86)/Microsoft Visual Studio/2019/Community/VC/Tools/MSVC/x64_24.28314/bin/mstools64/x64/cl.exe  
Check for working CXX compiler: C:/Program Files (x86)/Microsoft Visual Studio/2019/Community/VC/Tools/MSVC/x64_24.28314/bin/mstools64/x64/cxxcl.exe -- works  
Detecting CXX compiler ABI info  
Detecting CXX compiler ABI info - done  
Detecting CXX compile features  
Detecting CXX compile features - done  
Using [NVCC] compiler  
Looking for asprintf  
Looking for asprintf - not found  
Looking for reallocarray  
Looking for reallocarray - not found  
Looking for strcasestr  
Looking for strcasestr - not found  
Looking for strtcat  
Looking for strtcat - not found  
Looking for strtolcpy  
Looking for strtolcpy - not found  
Looking for strtokdup  
Looking for strtokdup - not found  
Looking for strstr  
Looking for strstr - not found  
Looking for timesg  
Looking for timesg - not found  
Looking for arc4random_buf  
Looking for arc4random_buf - not found  
Looking for arc4random_uniform  
Looking for arc4random_uniform - not found  
Looking for explicit_bzero  
Looking for explicit_bzero - not found  
Looking for getauxval  
Looking for getauxval - not found  
Looking for getentropy  
Looking for getentropy - not found  
Looking for getpagesize  
Looking for getpagesize - not found  
Looking for timingsafe_bcmp  
Looking for timingsafe_bcmp - not found  
Looking for timingsafe_memcmp  
Looking for timingsafe_memcmp - not found  
Looking for memmem  
Looking for memmem - not found  
Looking for include file err.h  
Looking for include file err.h - not found  
Looking for clock_gettime  
Looking for clock_gettime - not found  
Looking for sys/types.h  
Looking for sys/types.h - found  
Looking for stdint.h  
Looking for stdint.h - found  
Looking for stddev.h  
Looking for stddev.h - found
```

After completion of the previous command, now ask for cmake to build with the following command:

```
"C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE\CommonExtensions\Microsoft\CMake\CMa
ke\bin\cmake.exe" --build . --config Release
```

The program should now compile. It will take some time.

```
C:\Users\Eron\Desktop\Neuman\Neuman\build\C:\Program Files (x86)\Microsoft Visual Studio 2019 Community\ToolsCommonExtensions\VisualStudio\Microsoft\Make\Make.exe" --build -.-config Release
Microsoft® Build Engine version 16.4.0+e901037fe for .NET Framework
Copyright (C) Microsoft Corporation. All rights reserved.

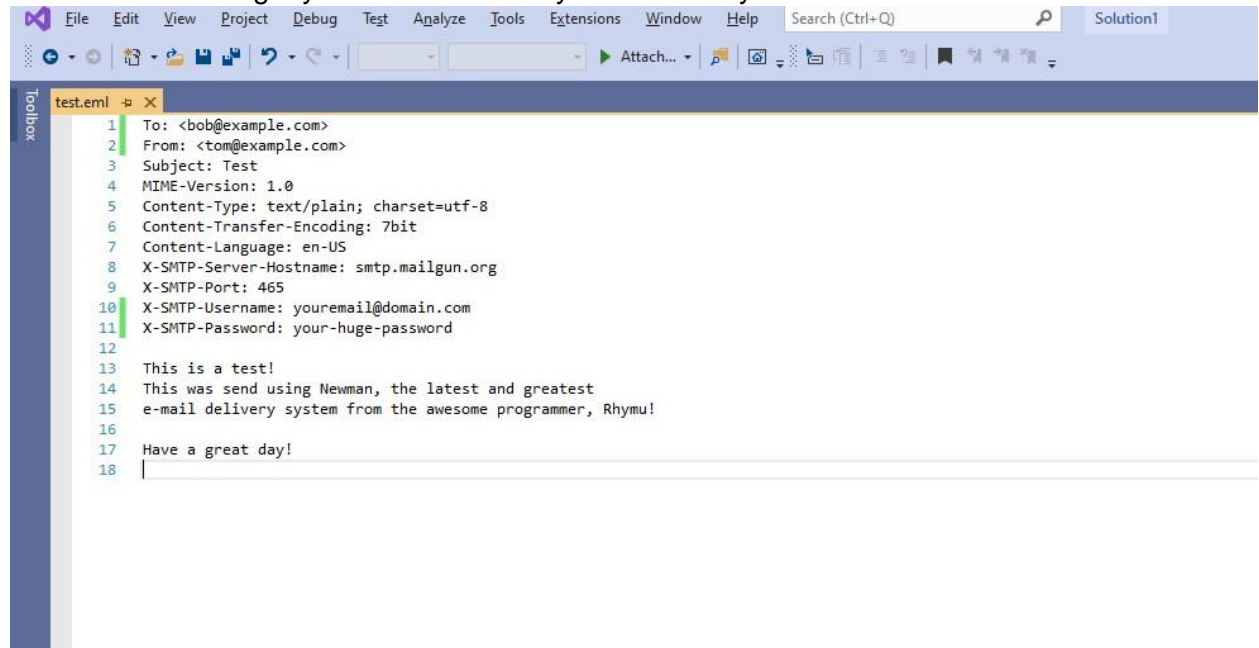
Checking Build System
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/Base64/CMakeLists.txt
Base32.cpp
Base64.cpp
Generating Code...
Base64.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/Base64/Release/Base64.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/HASH/CMakeLists.txt
Hmac.cpp
Hotp.cpp
Mds.cpp
Sha256.cpp
Sha1.cpp
Sha2.cpp
Totp.cpp
Generating Code...
Hash.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/HASH/Release/Hash.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/StringExtensions/CMakeLists.txt
StringExtensions.cpp
C:/Users/Eron/Desktop/Neuman/Neuman/StringExtensions/src/StringExtensions.cpp(45,5): warning C4996: 'wctombs': This fu
may be unsafe. Consider using wctombs_s instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help
for details. [C:/Users/Eron/Desktop/Neuman/Neuman/build/StringExtensions/StringExtensions.vcxproj]
C:/Program Files (x86)/Windows Kits/10/Include/10.0.18362.0/ucrt/stdlib.h(1015): message : see declaration of 'wctombs'
[C:/Users/Eron/Desktop/Neuman/Neuman/build/StringExtensions/StringExtensions.vcxproj]
StringExtensions.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/StringExtensions/Release/StringExtensions.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/MessageHeaders/CMakeLists.txt
MessageHeaders.cpp
MessageHeaders.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/MessageHeaders/Release/MessageHeaders.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/SystemAbstractions/CMakeLists.txt
DataQueue.cpp
DiagnosticsContext.cpp
DiagnosticsSender.cpp
DiagnosticsStreamReporter.cpp
File.cpp
NetworkConnection.cpp
NetworkEndpoint.cpp
Stringfile.cpp
ClipboardWin32.cpp
CryptRandomWin32.cpp
DirectoryMonitorWin32.cpp
DynamicLibraryWin32.cpp
FileWin32.cpp
NetworkConnectionWin32.cpp
NetworkEndpointWin32.cpp
ServiceWin32.cpp
SubprocessWin32.cpp
TargetInfoWin32.cpp
TimeWin32.cpp
Generating Code...
SystemAbstractions.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/SystemAbstractions/Release/SystemAbstractions
.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/Sasl/CMakeLists.txt
Plain.cpp
Login.cpp
Scram.cpp
C:/Users/Eron/Desktop/Neuman/Neuman/Sasl/src/ClientScram.cpp(370,5): warning C4996: 'scanf': This function or variab
le may be unsafe. Consider using scanf_s instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help
for details. [C:/Users/Eron/Desktop/Neuman/Neuman/build/Sasl/Sasl.vcxproj]
C:/Program Files (x86)/Windows Kits/10/Include/10.0.18362.0/ucrt/stdio.h(2263): message : see declaration of 'scanf' [
C:/Users/Eron/Desktop/Neuman/Neuman/build/Sasl/Sasl.vcxproj]
Generating Code...
Sasl.vcxproj -> C:/Users/Eron/Desktop/Neuman/Neuman/build/Sasl/Release/Sasl.lib
Building Custom Rule C:/Users/Eron/Desktop/Neuman/Neuman/SmtP/CMakeLists.txt
```

6. Using the program

To use the program, you need both an .eml file containing the e-mail info and a .pem file containing an SSL certificate.

The .pem file can be found in Newman/LibreSSL/apps/openssl/cert.pem

The .eml file comes in Newman main folder, but some changes must be made. Also, you need an account in a working SMTP server. I recommend Mailgun.com. Create and account there and get your credentials so you can modify the .eml file as below.

A screenshot of a code editor window titled 'test.eml'. The editor has a menu bar with 'File', 'Edit', 'View', 'Project', 'Debug', 'Test', 'Analyze', 'Tools', 'Extensions', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The code is as follows:

```
1 To: <bob@example.com>
2 From: <tom@example.com>
3 Subject: Test
4 MIME-Version: 1.0
5 Content-Type: text/plain; charset=utf-8
6 Content-Transfer-Encoding: 7bit
7 Content-Language: en-US
8 X-SMTP-Server-Hostname: smtp.mailgun.org
9 X-SMTP-Port: 465
10 X-SMTP-Username: youremail@domain.com
11 X-SMTP-Password: your-huge-password
12
13 This is a test!
14 This was send using Newman, the latest and greatest
15 e-mail delivery system from the awesome programmer, Rhymu!
16
17 Have a great day!
18 |
```

Move both files to Newman\build\Release.

Then, all you need to do is go to the command line, navigate to folder Newman\build\Release and type the command:

Newman email.eml cert.pem

```
C:\Users\Eron\Desktop\Newman-master\Newman\build\Release>Newman.exe test.eml cert.pem
[0.003393 Newman:3] Connecting to SMTP server.
[0.178077 Newman:3] Connected to SMTP server.
[0.178867 Newman:3] Preparing to send e-mail...
[0.475766 Smtplib:1] S: 220 ak47 ESMTP ready
[0.476390 Smtplib:1] C: EHLO [192.168.15.12]
[0.622667 Smtplib:1] S: 250-ak47
[0.623151 Smtplib:1] S: 250-AUTH PLAIN LOGIN
[0.623968 Smtplib:1] S: 250-SIZE 52428800
[0.624697 Smtplib:1] S: 250-8BITMIME
[0.625369 Smtplib:1] S: 250-ENHANCEDSTATUSCODES
[0.626121 Smtplib:1] S: 250 SMTPUTF8
[0.626765 SmtplibAuth:0] Plain: C: AUTH PLAIN \0edauricio@cfdyn.com\0*****
[0.797975 SmtplibAuth:0] S: 235 2.0.0 OK
[0.798487 Newman:3] Sending e-mail.
[0.799264 Smtplib:1] C: MAIL FROM:<edauricio@cfdyn.com>
[0.800137 Newman:3] Waiting for e-mail to be sent...
[0.946147 Smtplib:1] S: 250 Sender address accepted
[0.946781 Smtplib:1] C: RCPT TO:<eron.tiago90@gmail.com>
[1.093717 Smtplib:1] S: 250 Recipient address accepted
[1.094482 Smtplib:1] C: DATA
[1.240975 Smtplib:1] S: 354 Continue
[1.241469 Smtplib:1] C: To: <eron.tiago90@gmail.com>
From: <edauricio@cfdyn.com>
Subject: Test
MIME-Version: 1.0
Content-Type: text/plain; charset=utf-8
Content-Transfer-Encoding: 7bit
Content-Language: en-US

[1.246084 Smtplib:1] C: This is a test!
This was send using Newman, the latest and greatest
e-mail delivery system from the awesome programmer, Rhymu!

Have a great day!
[1.249332 Smtplib:1] C: .
[1.640589 Smtplib:1] S: 250 Great success
[1.641175 Newman:3] E-mail successfully sent.
[1.642118 Newman:3] Exiting...
[1.642922 Smtplib:1] TlsDecorator: NetworkConnection: closed connection
[1.643789 Smtplib:1] TlsDecorator: NetworkConnection: connection closed abruptly by peer

C:\Users\Eron\Desktop\Newman-master\Newman\build\Release>
```