Subject: Pop3 class and reference examples for U++
Posted by Oblivion on Fri, 17 Jan 2014 00:49:06 GMT

View Forum Message <> Reply to Message

Hello,

I'ts been a long time since I upload some code to bazaar and I would like to upload my brand new Pop3 class (and reference examples) to the bazaar if the admins (Koldo?) could grant me the SVN/bazaar access.

Pop3 is a Upp::TcpSocket derived POP3 encapsulation class, very much like the Upp::Smtp class in its public api.
While it conforms to the RFC 1939, it is implemented with simplicity in mind. Currently, it is in beta status. While its public api is stable, internals will probably change, for I am planning to add Capabilities (CAPA command) support. So there is of couse room for improvements, such as writing a much flexible and Upp/C++ friendly parser to substitute sscanf command, fixing bugs, refactoring the code, etc.

I tried to test it as extensively as I can, with different and popular POP3 providers.

And I wrote a simple console based reference example (again, similar to the SMTP reference example), which can retrieve message headers. I am also writing a GUI based one. I will upload it here in the next days.

Pop3 class requires OpenSSL. If you are using Linux, you probably have it. If you are Using windows, you can either get the source and compile it, or better, simply download the precompiled binaries from: (http://www.openssl.org/related/binaries.html) and configure it using TheIDE's "Setup"->"Build Methods..." menu.

Pop3 class and relevant example code are tested under:

U++ version 6738

Arch Linux: Linux 3.12 i686 Kernel with KDE SC 4.12.
Windows XP SP3 (i686)

Any suggestions, bug reports are always welcome.

And a happy new year to everyone!

P.s.: There are two helper functions in Pop3Example reference code: One ise DecodeEncodedString(): to decode QuotedPrintable/Q encoded texts. And the other is FindHeadersElement(): to get header sections. I wrote them in a hurry and long time ago so they are not meant to stay. I am planning to write an InternetMessage/EMail class, so they will be replaced with proper and fresh methods.

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by koldo on Fri, 17 Jan 2014 07:12:01 GMT
View Forum Message <> Reply to Message

Hello Oblivion

Please check this link about OpenSSL installing if it is good for you.

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Fri, 17 Jan 2014 07:27:53 GMT
View Forum Message <> Reply to Message

Looks good, except

sscanf(line, "%s %s", &tag1[0], &tag2[0]);

lines... (using sscanf with fixed buffers is security problem, also instead of &tag1[0] you can write
~tag1).

With your permission, I would do CR to it, fix some hard edges and put into uppsrc as
plugin/POP3.

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Fri, 17 Jan 2014 11:00:30 GMT
View Forum Message <> Reply to Message

Quote:
Looks good, except

sscanf(line, "%s %s", &tag1[0], &tag2[0]);

lines... (using sscanf with fixed buffers is security problem, also instead of &tag1[0] you can write
~tag1).

With your permission, I would do CR to it, fix some hard edges and put into uppsrc as

plugin/POP3.

Hello mirek.

Formally: permission granted. Informally: Of course, please feel free correct it, fix it and put it into uppsrc. I would be happy to contribute to U++ anyways Also I am working on a Ftp class, so any corrections on or fine-tuning of this code will make the Ftp class and my understanding of the TcpSocket class better.

By the way, I have two questions:

1. Do you allow patches? I was planning to add CAPA command and replace the sscanf with a flexible parser. If you are going to add these, fine. If not, I would like to work on it.

2. This is somewhat a general question. Do you have any future plans to add a Scan() command (like the Format() command) to U++?

Quote:
Hello Oblivion

Please check this link about OpenSSL installing if it is good for you.

Thanks for the information, koldo!

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Tue, 25 Mar 2014 09:07:14 GMT
View Forum Message <> Reply to Message

I have done some minor changes and added class InetMessage to simplify parsing. Otherwise, it is now Core/POP3, reference/Pop3. Thanks a lot, this is very helpful.

Quote:
2. This is somewhat a general question. Do you have any future plans to add a Scan() command (like the Format() command) to U++?

Usually I prefer CParser. Anyway, your use of sscanf inspired me to introduce SplitTo function....

Mirek

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Wed, 26 Mar 2014 09:34:47 GMT

I believe that your original code for multiline responses is wrong (found that hard way on actual message).

I have changed the code to

```
  for(;;) {
   line = GetDataLine();
   if(line.IsEmpty())
    break;
   if(line == ".\r\n") {
    LLOG("<< ...");
    return true;
   }
   if(*line == '.')
    data << line.Mid(1);
   else
    data << line;
  }
```

Please check...

Mirek

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Wed, 26 Mar 2014 11:37:37 GMT

Hello Mirek,

Ah.. I didn't encounter any actual errors but yes, looking more closely, you are right that was my fault, sorry. And thanks for correcting it.

Other than that, after furter examining the code, I shall report an error in the Pop3::GetListItems() method:
Current code omits the last line of the list. So I corrected it.

```
bool Pop3::GetListItems(ValueMap& list, dword type1, dword type2)
{
 StringStream s(data);
```

```
      for(;;) {
  String line = s.GetLine();
  if(s.IsError())
   return false;
  if(s.IsEof())                        // << ---- Omits last line of the list. EOF checking should be
moved to top.
   break;
  Vector<String> s = Split(line, ' ');
  if(s.GetCount() < 2)
   return false;
  list.Add(Scan(type1, s[0]), Scan(type2, s[1]));
 }
 return true;
}
```

Corrected version:

```
bool Pop3::GetListItems(ValueMap& list, dword type1, dword type2)
{
 StringStream s(data);
 while(!s.IsEof()) {
  String line = s.GetLine();
  if(s.IsError())
   return false;
  Vector<String> s = Split(line, ' ');
  if(s.GetCount() < 2)
   return false;
  list.Add(Scan(type1, s[0]), Scan(type2, s[1]));
 }
 return true;
}
```

Other than that, everything seems fine.

P.S. Thanks a lot for the SplitTo() function, it is and will be very helpful. Too bad those handy
utility functions got undocumented.
When I have time I'll examine and document them.

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Wed, 26 Mar 2014 17:31:10 GMT

Also I propose a change:
I was looking into the possibility of using TcpSocket::GetLine() instead of Pop3::GetDataLine()
method and come up with the below modification.
You are the expert, so please comment if this is valid. (It seems so and works, but again, I am
suspicious).

```cpp
bool Pop3::PutGet(const String& s, bool multiline, bool nolog)
{

    // Put() request.
    if(!s.IsEmpty()) {
        if(!nolog)
            LLOG(">> " << TrimRight(s));
        if(!PutAll(s)) {
            LLOG("-- " << GetLastError());
            return false;
        }
    }
    // Get() response.
    data.Clear();
    String line = GetLine();
    if(!line.IsVoid()) {
        LLOG("<< " << line);
        if(line.StartsWith("+OK")) {
            if(!multiline) {
                data.Cat(line);
                return true;
            }
            else
                for(;;) {
                    line = GetLine();
                    if(line.IsVoid()) // IsEmpty() cannot be used here.
                        break;
                    if(line == ".") {
                        LLOG("<< ...");
                        return true;
                    }
                    data.Cat(*line == '.' ? line.Mid(1) : line);
                    data.Cat("\r\n");
                }
        }
        else
        if(line.StartsWith("-ERR"))
            error = line;
    }
```

```
        LLOG("-- " << GetLastError());
        return false;

}
```

Basically, it appends the CRLF after downloading. This way, we can discard GetDataLine() method in favor of native TcpSocket method.

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Thu, 27 Mar 2014 07:45:15 GMT
View Forum Message <> Reply to Message

Yeah, that is better.

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Thu, 27 Mar 2014 14:24:52 GMT
View Forum Message <> Reply to Message

Hello Mirek,

Latest revision of Pop3 has a constant value (int LINE_MAX) used in GetLine(). But afaik, LINE_MAX is actually a posix macro used by gcc. So it gives an error:

../Ultimate++/uppsrc/Core/POP3/POP3.cpp: In member function 'bool Pop3::PutGet(const Upp::String&, bool, bool)':
..Ultimate++/uppsrc/Core/POP3/POP3.cpp:114:12: error: expected unqualified-id before numeric constant
  const int LINE_MAX = 20000000;
        ^

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Thu, 27 Mar 2014 14:38:12 GMT
View Forum Message <> Reply to Message

Fixed. I have also hugely refactored InetMessage....

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Sun, 30 Mar 2014 00:48:57 GMT

Hello Mirek,

I refactored the DecodeHeaderValue() helper function of InetMessage.
The old code was written hastily and supplied for demonstration purpose only; it had some problems, so it shouldn't be used.
I attached the refactored version (tested with my several mailboxes), I would be grateful if you could review it.

Regards.

```
File Attachments
1) InetMessage.cpp, downloaded 382 times
```

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Sun, 30 Mar 2014 07:29:51 GMT

Thanks, I was thinking about this too.

Mirek

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Wed, 09 Apr 2014 23:24:58 GMT

Hello Mirek,

I've noticed that you refactored the INetMessage and SMTP classes. When I looked into the code of Smtp class I noticed that the quoted-printable encoding was hard coded into the class.
So I took the opportunity to improvise on the existing codebase. While I did not touch the Smtp class, but I wrote a generic QPEncode() function to accompany the existing QPDecode() function.
There seem to be two problems with the current QP encoding of Smtp (if I am not missing anything, or got something totally wrong :)):

1) It violates the maxlen (should be <= 76 chars) of quoted-printable text (RFC 1521, rule 5).
2) It does not encode horizontal tab (=09) or space (=20) at the EOL, just before the CRLF (it should). (This statement is wrong, it turns out that I misread the code, it was only implicit. :blush:)

So, I created the function and patched accordingly the source files (inet.h, inet.cpp) and the documentation (inet$en-us.tpp)

I tested it with a lot of strings, but as usual, it always needs to be tested more. Please find enclosed the patched files.
As usual, I would be grateful if you could please review it.

EDIT: Fixed misinformation.

Regards.

File Attachments
1) QPEncode_Patch.tar.gz, downloaded 291 times

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Thu, 10 Apr 2014 14:14:53 GMT

Hello Mirek,
I updated the QPEncode() function using the original code snippet in SMTP. It turned out that the coversion of HT and SP was implicitly present, I guess I should read more carefully next time. Anyways, my previous version of QPEncode was working but this version is more compact (unnecessary code removed) and also encodes the last space character right before the EOF to QP, if no CRLF present at the end.

Regards.

File Attachments
1) InetUtil.cpp, downloaded 442 times

Subject: Re: Pop3 class and reference examples for U++
Posted by Oblivion on Wed, 16 Apr 2014 01:00:11 GMT

Hello Mirek,
I see that you recently added ParseMessageIDs() method to the InetMessage class.
The code block in that function is something I encounter often. Especially when parsing some text (you can even see it in the Pop3::GetTimestamp()).
Maybe we should put that code under String utilities and change the code accordingly?
So I wrote the following Slice() functions which you can also come by in some other C++/Java frameworks.

```
int Slice(const String& s, String& d, const String& delim1, const String& delim2, int pos = 0)
{
 int b = -1, e = -1;
 if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, b += delim1.GetLength())) == -1)
```

```
  return -1;
 d = s.Mid(b, e - b);
 return e += delim2.GetLength();
}

int Slice(const WString& s, WString& d, const WString& delim1, const WString& delim2, int pos =
0)
{
 int b = -1, e = -1;
 if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, b += delim1.GetLength())) == -1)
  return -1;
 d = s.Mid(b, e - b);
 return e += delim2.GetLength();
}

int Slice(const String& s, String& d, int delim1, int delim2, int pos = 0)
{
 int b = -1, e = -1;
 if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, ++b)) == -1)
  return -1;
 d = s.Mid(b, e - b);
 return ++e;
}

int Slice(const WString& s, WString& d, int delim1, int delim2, int pos = 0)
{
 int b = -1, e = -1;
 if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, ++b)) == -1)
  return -1;
 d = s.Mid(b, e - b);
 return ++e;
}

String Slice(const String& s, const String& delim1, const String& delim2)
{
 String r;
 return (Slice(s, r, delim1, delim2) == -1) ? String::GetVoid() : r;
}

WString Slice(const WString& s, const WString& delim1, const WString& delim2)
{
 WString r;
 return (Slice(s, r, delim1, delim2) == -1) ? WString::GetVoid() : r;
}

String Slice(const String& s, int delim1, int delim2)
{
 String r;
```

```
  return (Slice(s, r, delim1, delim2) == -1) ? String::GetVoid() : r;
}

WString Slice(const WString& s, int delim1, int delim2)
{
 WString r;
 return (Slice(s, r, delim1, delim2) == -1) ? WString::GetVoid() : r;

}
```

They are really useful for parsing text. I already patched my local copy of upp (with its api reference doc in "String utility functions" section).
By the way, I can upload the patched files if you have time to review them and also approve them.

Regards.

Subject: Re: Pop3 class and reference examples for U++
Posted by mirek on Mon, 28 Apr 2014 15:48:21 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Thu, 10 April 2014 16:14Hello Mirek,
I updated the QPEncode() function using the original code snippet in SMTP. It turned out that the coversion of HT and SP was implicitly present, I guess I should read more carefully next time. Anyways, my previous version of QPEncode was working but this version is more compact (unnecessary code removed) and also encodes the last space character right before the EOF to QP, if no CRLF present at the end.

Regards.

Thanks, applied.

Mirek