

## Re: C# Equivalent of C++ MD5 Algorithm

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2006-11/msg04053.html>

---

- *From:* [galtschul@xxxxxxxxxx](mailto:galtschul@xxxxxxxxxx)
  - *Date:* 21 Nov 2006 15:33:21 -0800
- 

I've looked at that post and it's basically doing exactly what I'm doing, The only difference is tthat the author treats the incoming string as unicode (which mine is not. It's ascii) and he converts to HEX (which I can't do.. I have to convert to base64 since that's how it's stored in the database). I did try this approach and, although I got a different hash returned, it contained the same amount number of charectars.

I don't see any reference here to ASN.1 encoding of a PKCS #7 X.509 message. How do I go about doing this in C#?

Regards,

Gregg

Peter wrote:

Here is somebody's blog post that should point you in the right direction.  
It's not as complicated as you indicated:

<http://blog.stevex.net/index.php/c-code-snippet-creating-an-md5-hash-string/>

Peter

--

Co-founder, Eggheadcafe.com developer portal:

<http://www.eggheadcafe.com>

UnBlog:

<http://petesbloggerama.blogspot.com>

"galtschul@xxxxxxxxxx" wrote:

How do I go about implementing that in C#. I tried some of the various x509 and pkcs7 classes and I kept getting exceptions having to do with

## Re: C# Equivalent of C++ MD5 Algorithm

certificates and whatnot. I felt like I was wasting my time going down the wrong path and this stuff doesn't seem too well documented on msdn.microsoft.com.

Regards

Ben Voigt wrote:

<galtschul@xxxxxxxxxx> wrote in message  
[news:1164141343.043155.186300@xx](mailto:news:1164141343.043155.186300@xx)

I am re-writing a C++ application in C# that takes a user's password, encrypts it using MD5 (I think), and compares it to what was encrypted and stored in the database when the user initially created their password. The problem is that the C++ encryption generates 110 characters and the C# encryption generates only 24. The interesting thing is that the 24 characters generated by the C# MD5 algorithm matches the last 24 characters of the C++ encryption algorithm. Here is the C++ code:

The C++ code is requesting ASN.1 encoding of a PKCS #7 X.509 message, which will naturally add a humonguous header to the data.

```
/*  
std::string  
PasswordHash::get_passwordhash(char *s)  
{  
std::string hashed_value;  
  
const BYTE* sval[1];  
unsigned long lval[1];  
  
sval[0] = reinterpret_cast<BYTE*>(s);  
lval[0] = strlen(s);  
  
CRYPT_ALGORITHM_IDENTIFIER  
AlgId;  
AlgId.pszObjId=szOID_RSA_MD5;  
AlgId.Parameters.cbData=0;
```

Re: C# Equivalent of C++ MD5 Algorithm

```
CRYPT_HASH_MESSAGE_PARA hash;
hash.cbSize =
sizeof(CRYPT_HASH_MESSAGE_PARA);
hash.dwMsgEncodingType =
(PKCS_7_ASN_ENCODING |
X509_ASN_ENCODING);
hash.hCryptProv = NULL;
hash.HashAlgorithm = AlgId;
hash.pvHashAuxInfo = NULL;

unsigned long hash_length = 0;
if(!CryptHashMessage( &hash, FALSE, 1,
sval, lval, NULL,
&hash_length, NULL, NULL))
{
DWORD error = GetLastError();
return hashed_value;
}

BYTE* hash_data = new
BYTE[hash_length + 1];
ZeroMemory(hash_data, hash_length + 1);

if(!CryptHashMessage(&hash, FALSE, 1,
sval, lval, hash_data,
&hash_length, NULL, NULL))
{
DWORD error = GetLastError();
return hashed_value;
}

std::vector<char > vhash_data;
vhash_data.resize(hash_length);
memcpy(&vhash_data[0], (void*)hash_data,
hash_length);

base64<char> encoder;
int state = 0;
encoder.put(vhash_data.begin(),
vhash_data.end(),
std::back_inserter(hashed_value), state,
base64<>::noline());
return hashed_value;
}

/*****
```

And here is the C# code that I'm using:

```
string generatePassword(string password)
{
```

## Re: C# Equivalent of C++ MD5 Algorithm

```
MD5 md5Hasher = new
MD5CryptoServiceProvider();
byte[] data =
md5Hasher.ComputeHash(Encoding.ASCII.GetBytes(password));
string s = Convert.ToBase64String(data);
return s;
}
```

### C++ OUTPUT

-----  
"MEcGCSqGSIb3DQEHBaA6MDgCAQAwdAYIKoZlhcNAgUFADATBgkqhkiG

### C# OUTPUT

-----  
"AETe7sQ97Rm5UhJQeesXgQ=="

Am I doing something wrong in the C#  
code? Perhaps I'm not fully  
understanding what the C++ code is doing.

Thanks in advance for any help I can get