

## Re: PasswordOption.SendNone

**Source:**

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices.enhancements/2004-05/>

---

**From:** Softwaremaker (*msdn\_at\_removethis\_softwaremaker.net*)

**Date:** 05/26/04

Date: Tue, 25 May 2004 19:51:07 -0700

Hello Graham,

Basically, the client sign or encrypt the HASH of the message (Digest) with the password to get the Ciphertext. The Message together with the Ciphertext is then sent over to the WS together with the username (No Password).

In the WSE Pipeline Filters, the WS will pull the password of the username (which is sent over from the client) from a database or whatever storage medium you have...(This model wont work if you query a LDAP AD) and then proceed to encrypt the HASH of the message (Digest) with the password to get the Ciphertext. Then proceed to compare the Ciphertext value to the one in the message. A user is authenticated once both Ciphertext matches.

I hope this clears up some of your confusion.

hth.

--

Thank you very much

Warmest Regards,

Softwaremaker

Architect | Evangelist | Consultant

Microsoft Regional Director

<http://www.microsoft.com/rd>

+++++

"Graham Allwood" <graham.allwood@nospamtoEborsolutions.com> wrote in message news:eEf4AULQEHA.1392@TK2MSFTNGP09.phx.gbl...

> Hi,

>> I am little confused with the PasswordOption enum when creating a UsernamePassword token. Is the following correct:

>> Using PasswordOption.SendHashed a hash of the password is sent with the message. At the server WSE uses a UsernameTokenManager derived class (typcially) to obtain a password it can hash and compare against the sent hashed password. The message can also be signed using the UsernamePassword token if required.

>> However, using PasswordOption.SendNone results in no password (or hash) being sent to the server. In that case how does the web service authenticate

> the message? Reading the docs it sounds to be that if you use this option > then you should also sign the message as the message is signed on the client

Re: PasswordOption.SendNone

microsoft.public.dotnet.framework.webservices.enhancements: Re: PasswordOption.SendNone

```
> using the password(??). Authentication is then achieved at the web server
by
> virtue of a password being required from the UsernameTokenManager to
verify
> the signed message.
>> Does this sound right?
>> I'm basically trying to establish whether I should be using SendHashed or
> SendNone. Can anyone enlighten me?
>>> TIA
>> Graham
>>> Using
>>
```