CodeBuildingBlocks.com

CBB AWSAuth – Amazon Web Services authentication fields generator

Technical paper v1.1

Introduction

CBB AWSAuth is an ActiveX component for generation of Amazon Web Services (AWS) authentication fields. The component contains only one COM object which is called "AWSAuth.AWSAuth". If you wish you can access the object by its GUID, "90CCB7E1-7397-4491-ADDF-6773A1E0CC10".

Features

- **Generation of the authentication fields required by AWS**, "Timestamp" and "Signature".
- **Generation of signatures for Amazon S3**. Sign your S3 canonical strings or any other data with a generic method.
- **Caching of cryptographic data to speedup multiple calls.** If you use the same secret key to generate many signatures you will benefit from this feature, unlike using some other methods which have no state and as a result they are forced to recalculate everything each time.
- **Caching of the signature-timestamp pair to speedup multiple calls.** If you use the same secret key and prefix to generate many signatures you will benefit from this feature. Amazon doesn't require your timestamp to be accurate to the second. You are allowed a certain time window for each authentication pair. AWSAuth will monitor your requests and will completely bypass new signature calculation when possible, as determined by the value of "TimestampTimeout" that you provide.
- **Optional URLEncoding of the returned signature.** Amazon requires the "Signature" parameter to be URLEncoded.

IAWSAuth interface

GenAuthentication()

A method for Alexa Web Information and Amazon Mechanical Turk services. GenAuthentication() returns the authentication pair, Signature-Timestamp. Prior to calling the method you have to set the SecretKey and ServiceOperationPrefix properties.

Script/VB syntax:
Signature=obj.GenAuthentication(byref Timestamp)

C++ syntax:
HRESULT GenAuthentication([out] VARIANT *Timestamp, [out, retval] VARIANT *Signature);

GenAuthenticationEx()

A generic method, mostly geared towards the Amazon S3 service. GenAuthenticationEx() returns the Signature of the String that is passed to it. Prior to calling the method you have to set the SecretKey property.

Script/VB syntax:
Signature=obj.GenAuthenticationEx(ContentString)

C++ syntax:
HRESULT GenAuthenticationEx([in] BSTR Content, [out, retval] VARIANT * Signature);

Properties

Name	Type	Description
SecretKey	BSTR	Set this property to the value of your secret key, as you received it from Amazon. Setting this property recalculates internal cryptographic data.
ServiceOperationPrefix	BSTR	Set this property to the value that should prefix the timestamp during the calculation of the signature. Normally this is Amazon Service name concatenated with Amazon Operation name (i.e., “AlexaWebInfoServiceUrlInfo”)
TimestampTimeout	long int	Specify how often (in seconds) to generate a new signature-timestamp pair when there is no change in the secret key or prefix. Set to 0 to disable this feature. The default value is 5 minutes.

		You can even stretch if further if you wish. Our tests have shown that Amazon accepts signatures which are up to 15 minutes old.
DoURLEncode	VARIANT_BOOL	Specify if you wish the returned signature to be URLEncoded or not. Amazon requires the "Signature" parameter to be URLEncoded; you may wish to disable this feature if you intend to encode the parameter yourself. The default value is enabled, TRUE.

Raw IAWSAuth interface data

```
interface IAWSAuth : IDispatch
{
[propget, id(1), helpstring("property SecretKey")]
HRESULT SecretKey([out, retval] BSTR *pVal);
[propput, id(1), helpstring("property SecretKey")]
HRESULT SecretKey([in] BSTR newVal);
[propget, id(2), helpstring("property ServiceOperationPrefix")]
HRESULT ServiceOperationPrefix([out, retval] BSTR *pVal);
[propput, id(2), helpstring("property ServiceOperationPrefix")]
HRESULT ServiceOperationPrefix([in] BSTR newVal);
[propget, id(3), helpstring("property TimestampTimeout")]
HRESULT TimestampTimeout([out, retval] long *pVal);
[propput, id(3), helpstring("property TimestampTimeout")]
HRESULT TimestampTimeout([in] long newVal);
[id(4), helpstring("method GenAuthentication")]
HRESULT GenAuthentication([out] VARIANT *Timestamp, [out, retval] VARIANT *Signature);
[propget, id(5), helpstring("property DoURLEncode")]
HRESULT DoURLEncode([out, retval] VARIANT_BOOL *pVal);
[propput, id(5), helpstring("property DoURLEncode")]
HRESULT DoURLEncode([in] VARIANT_BOOL newVal);
[id(6), helpstring("method GenAuthenticationEx")] HRESULT GenAuthenticationEx([in] BSTR
Content, [out, retval] VARIANT * Signature);
};
```

Support

If you require further technical support, please don't hesitate to contact us via support@codebuildingblocks.com. Also feel free to contact us with feature requests and suggestions.