

SRP_Encode

Encodes binary data into a printable format.

Syntax

```
Result = SRP_Encode(String, Algorithm, CharSet, Option)
```

Returns

The encoded data.

Parameters

Parameter	Description
String	The data to be encoded. Required.
Algorithm	The encoding algorithm to be used. Optional.
CharSet	The set of characters to be used for the encoder's output. Optional.
Options	An additional parameter for controlling the options of the given encoding algorithm. Optional.

Remarks

The SRP_Encode function is useful for encoding binary data into a printable format. A string that is in printable format contains human readable characters. Encoded data can usually be transmitted through the internet and used in URLs and emails. There are three encoding algorithms available:

Algorithm	Description	Version Added
HEX	The output contains hexadecimal pairs for each byte	
BASE32	The output is encoded in BASE32 printable format	
BASE64	The output is encoded in BASE64 printable format	
URL	The output is encoded in a format suitable for use in a URL	2.1.5
BASE64URL	The output is encoded in BASE64 using a character set suitable for use in a URL	2.1.5

If you leave the Algorithm parameter blank or pass an unrecognized value, then BASE64 is assumed.

OpenInsight 7.1 and later includes a Base64Encode function. The output will be the same as SRP_Encode, though the C++ algorithms used by the SRPUtilities.dll will likely perform faster. This is negligible for small amounts of data. SRP_Encode will work for OpenInsight 7.0 and earlier.

OpenInsight also includes it's own "HEX" encoder via the OConv function. However, this encoder does not encode delimiters. This may be useful if you're encoding elements in an array. However, if you are encoding binary data, then the SRP_Encode "HEX" algorithm will serve more useful.

Use [SRP Decode](#) when you need to convert back to binary data.

The CharSet parameter allows you to customize the charset used for BASE32 or BASE64. Leaving this blank uses the standard character set recognized by most decoders, including the one built into OpenInsight. If you run into a scenario in which the built-in character set causes issues, then you can customize the character set to fit your needs. Note, however, that you will need to pass the exact same character set to [SRP Decode](#) in order to restore the data as intended.

The Options parameter is dependent upon the algorithm chosen. Currently, only the BASE64 algorithm supports a single option. By default, the BASE64 algorithm inserts CRLFs in the encoding per industry standards. Setting Options to 0 omits the CRLFs. [SRP_Decode](#) has no such Options parameter as it is smart enough to ignore CRLFs whether or not they are there.

Example

```
* encode some binary data using default (BASE64)
Data = \D1D2D3D4\
EncodedData = SRP_Encode(Data)
```

```
* HEX encode some binary data
Data = \D1D2D3D4\
EncodedData = SRP_Encode(Data, "HEX")
```

```
* BASE32 encode some binary data
Data = \D1D2D3D4\
EncodedData = SRP_Encode(Data, "BASE32")
```

```
* BASE64 encode some binary data
Data = \D1D2D3D4\
EncodedData = SRP_Encode(Data, "BASE64")
```