

Base64 接口使用说明

概念背景:

Base64是一种基于64个可打印字符来表示二进制数据的编解码方式。这种方法将每3个字节的二进制数据转化为4个Base64字符，从而实现数据的紧凑表示和跨不同平台和协议的传输。

这种编码方式在网络上广泛应用于传输较长的标识信息，以及在HTTP环境下传递二进制数据，因为它可以增加数据的可读性和可移植性。同时，由于其不可读性，Base64编码也被用于一些安全措施中，例如隐藏一些机密信息在公开的文本中。

需要在项目中包含的文件

```
libbase64.h      #base64基础接口头文件
CustomOutLog.h   #定制输出日志接口(将FileToBase64中的日志输出与当前项目所用日志输出衔接)
FileToBase64.h   #对base64基础接口的封装,实现对文件进行加解码操作

libbase64.c
FileToBase64.cpp
```

常用接口使用说明

在代码中 `#include "FileToBase64.h"`

1. 对字符串进行加密(变成base64码)

```
//函数原型为:
void BASE64_EXPORT base64_encode
( const char    *src
, size_t       srclen
, char         *out
, size_t       *outlen
, int          flags
) ;
```

参数说明:

src: 需要加密的字符串
srclen: 需要加密的字符串的长度
out : 输出缓冲区
outlen: 输出缓冲区的大小(且大小需要大于输入src长度的4/3, 建议申请空间为 (strlen(src) * (4/3)) + 16)
flags : 默认设置成0

2. 对base64码字符串进行解码(将base64码变成正常文本)

```
//函数原型为:
int BASE64_EXPORT base64_decode
( const char    *src
, size_t       srclen
, char         *out
, size_t       *outlen
, int          flags
) ;
```

参数说明:

src: 需要解密的字符串
srclen: 需要解密的字符串的长度
out : 输出缓冲区
outlen: 输出缓冲区的大小(且大小需要大于输入src长度的3/4)
flags : 默认设置成0

返回值:

1 成功
0 失败

3. 对文件进行加密(变成base64码) 或 解码(将base64码变成正常文本)

接口封装在C_FB64类中,使用时需要生成对象再使用

1. 对文件进行加解密操作,在生成输出文件时默认情况下是所有文件内容生成在一行中(这也是通用的做法,这种格式可以与外部第三方接口通用;)
2. 但此接口支持对文件加解密操作生成输出文件时设置一行的字符数,如果内容中超过一行的字符数则换行(这种格式不通用,只能在自研的加解密接口中使用,且要求加解密过程中wrap值不能进行变动,即加密与解密码时需要wrap值保持不变,否则可能解析异常).

设置wrap的接口如下:

```
//默认不设置时为0, 输出格式不换行
C_FB64::C_FB64();
C_FB64::C_FB64(size_t wrap);

void C_FB64::set_wrap(size_t wrap);
```

文件加密接口

```
//函数原型为:
int C_FB64::encode_file(const char* file, const char *outfile=NULL);
```

参数与返回值说明:

```
1.如果outfile为NULL
   将文件 file 的内容转换成base64 存储为新文件file.encode
2.如果outfile不为NULL
   将文件 file 的内容转换成base64 存储为新文件outfile

返回值: 0 sucess
```

文件解码接口

```
//函数原型为:
int  C_FB64::decode_file(const char* file, const char *outfile=NULL);
```

参数与返回值说明:

```
1.如果outfile为NULL
   将文件内容为base64的文件file 的内容decode成原内容存储为新文件file.decode
2.如果outfile不为NULL
   将文件内容为base64的文件file 的内容decode成原内容存储为新文件outfile

返回值: 0 sucess
```