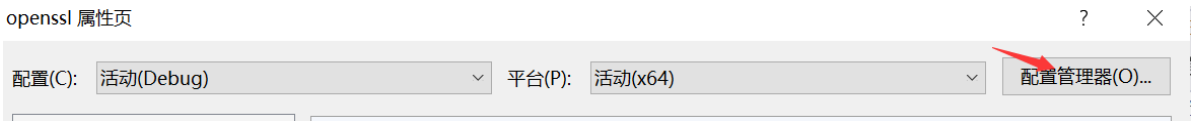
openssl 配置方法 （visual studio 2019）

1、下载安装openssl(下载X64版本)

2、打开visual studio

a、创建一个新项目

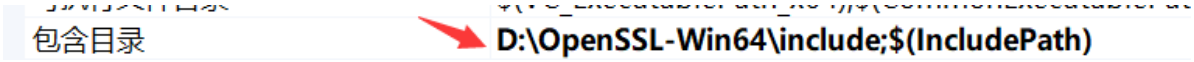
b、项目->属性->配置管理器



将默认的win32平台修改为下X64

c、项目->属性->VC++目录

在包含目录中添加openssl的include文件



在库目录中添加openssl的lib文件

d、项目->属性->链接器->输入->附加依赖项

键盘输入：

```
libcrypto.lib
libssl.lib
```

e、项目->属性->c/c++->->预处理器->预处理项定义->编辑

在最后输入

```
_CRT_SECURE_NO_WARNINGS
```

f、最后输入实例代码测试

```
extern "C"
{
#include <openssl/applink.c>
};
#include <stdio.h>
#include <openssl/evp.h>
#include <openssl/bio.h>
int main(void)
{
    EVP_MD_CTX* ctx = NULL;
    EVP_MD* sha256 = NULL;
    const unsigned char msg[] = {
        0x00, 0x01, 0x02, 0x03
    };
    unsigned int len = 0;
    unsigned char* outdigest = NULL;

```

```
    /* Create a context for the digest operation */
    ctx = EVP_MD_CTX_new();
    if (ctx == NULL)
        goto err;

    /*
     * Fetch the SHA256 algorithm implementation for doing the digest. We're
     * using the "default" library context here (first NULL parameter), and
     * we're not supplying any particular search criteria for our SHA256
     * implementation (second NULL parameter). Any SHA256 implementation
will
     * do.
     */
    sha256 = EVP_MD_fetch(NULL, "SHA256", NULL);
    if (sha256 == NULL)
        goto err;

    /* Initialise the digest operation */
    if (!EVP_DigestInit_ex(ctx, sha256, NULL))
        goto err;

    /*
     * Pass the message to be digested. This can be passed in over multiple
     * EVP_DigestUpdate calls if necessary
     */
    if (!EVP_DigestUpdate(ctx, msg, sizeof(msg)))
        goto err;

    /* Allocate the output buffer */
    outdigest = (unsigned char*)OPENSSL_malloc(EVP_MD_get_size(sha256));
    if (outdigest == NULL)
        goto err;

    /* Now calculate the digest itself */
    if (!EVP_DigestFinal_ex(ctx, outdigest, &len))
        goto err;

    /* Print out the digest result */
    BIO_dump_fp(stdout, outdigest, len);

err:
    /* Clean up all the resources we allocated */
    OPENSSL_free(outdigest);
    EVP_MD_free(sha256);
    EVP_MD_CTX_free(ctx);
}

```