

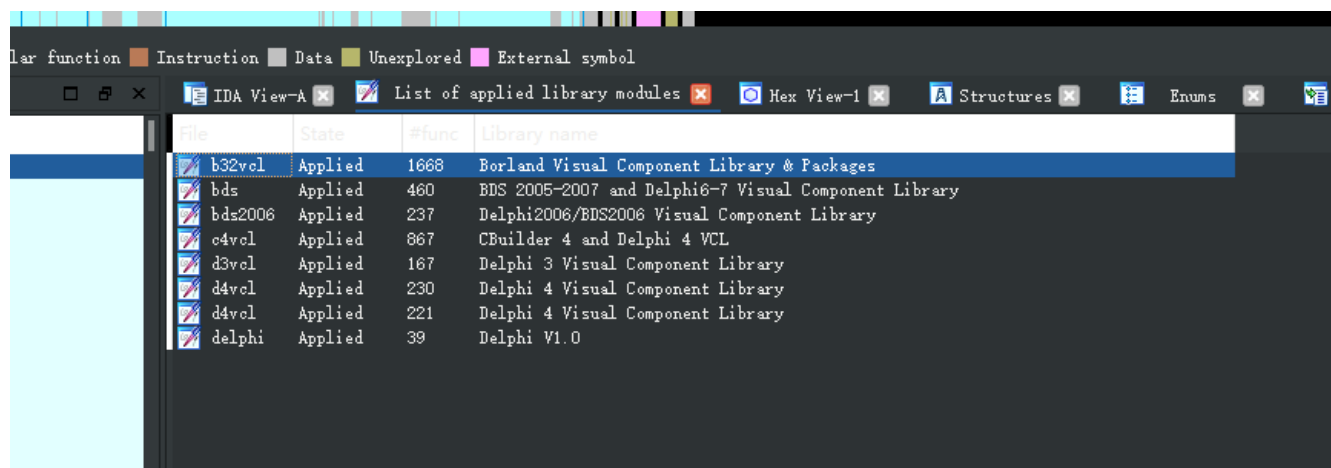
查壳
导入符号
分析程序
分析Cancella按钮点击事件
 分析关键函数
 写出注册机
 验证结果
分析ok按钮点击事件
 分析关键函数
 写出注册机
 校验结果

查壳



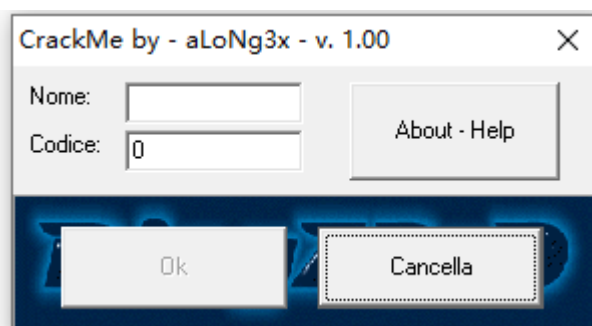
同样也是用Delphi写的，没有壳。

导入符号



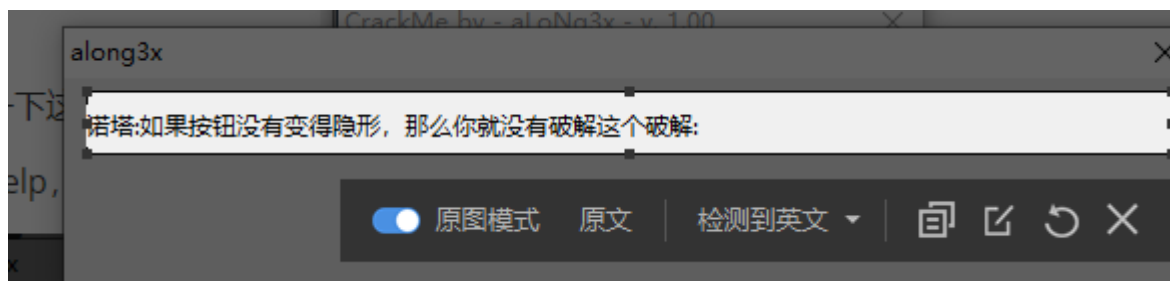
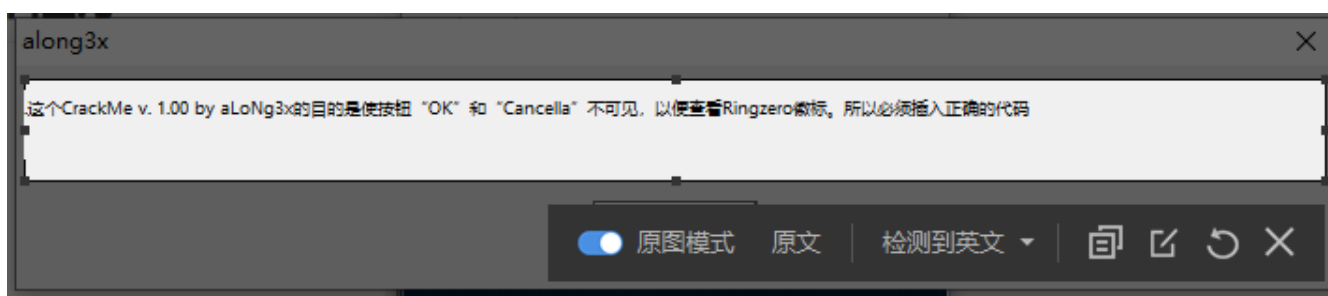
将程序载入到IDA，添加所有的Delphi的签名，然后导出为map文件，在OD中加载map文件，强大的签名库可以减少后面的分析时间。

分析程序



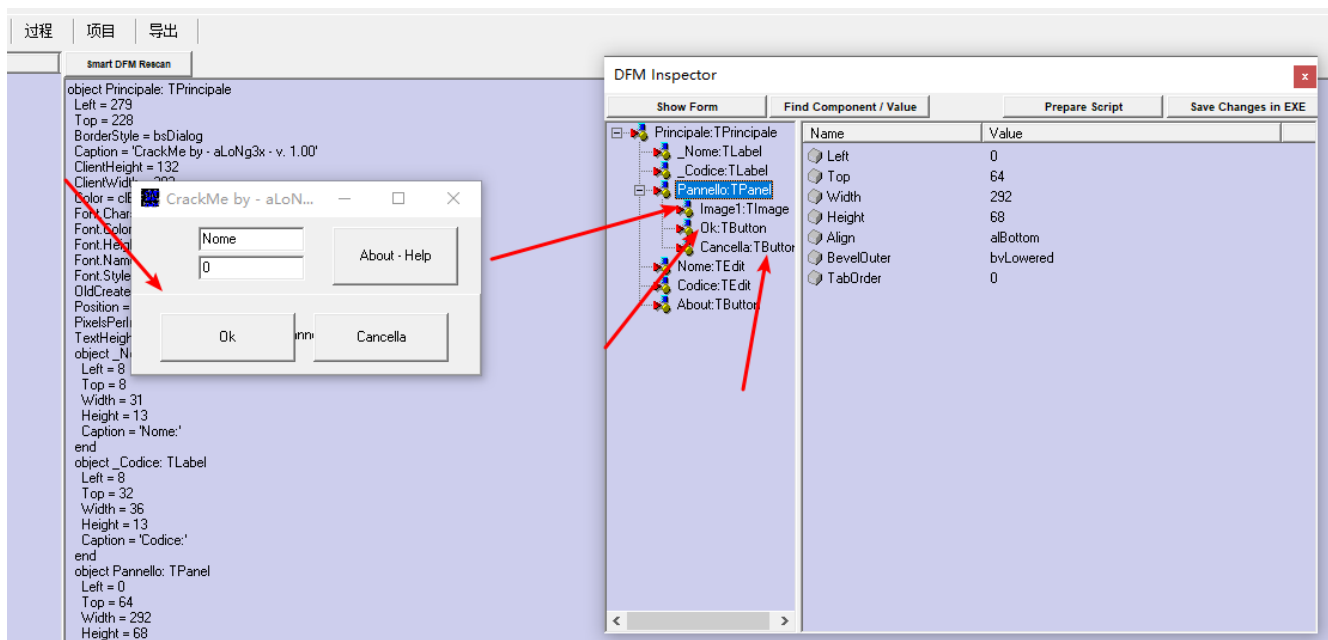
接下来分析一下这个程序，OK那个按钮被禁用了，这是什么套路？

旁边还有个help，不过都是英文的，上翻译。

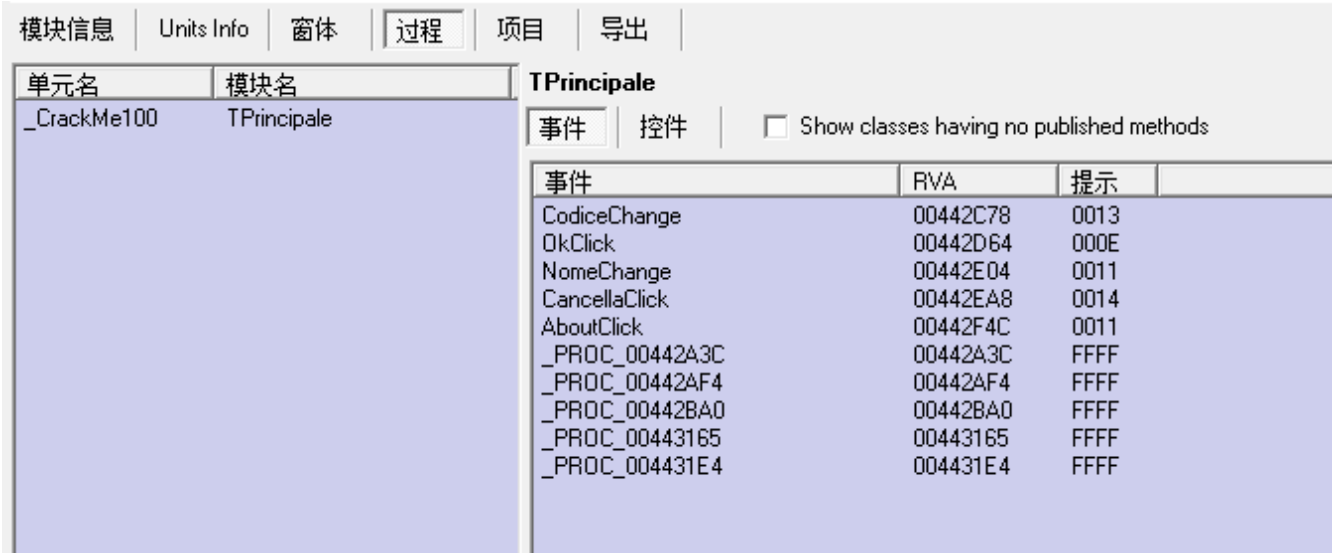


好，大概了解了。我们的目的就是要让那两个按钮变成隐藏的。

再拖到Darkde4里面查看一下按钮事件和窗口



从控件窗口可以看到在下面这个控件里有一个OK按钮 有一个Cancella按钮，还有一个图片。我们的目的应该就是要让那个图片显示出来。



事件按钮有如下几个响应事件，每个响应事件都有对应的RVA:

- 代码变换事件
- OK按钮点击事件
- 用户名变换事件
- Cancella按钮点击事件
- About按钮的点击事件

那么既然要让这两个按钮变的不可见，就从Cancella这个按钮的点击事件开始分析

分析Cancella按钮点击事件

找到Cancella按钮点击事件对应的RVA0x442EA8，随便输入一个账号密码，下断分析。

地址	HEX	数据
00442EA6	. C3	retb
00442EA7	. 90	nop
00442EA8	. 55	push ebp
00442EA9	. 8BEC	mov ebp, esp
00442EAB	. 6A 00	push 0x0
00442EAD	. 53	push ebx
00442EAE	. 8BD8	mov ebx, eax
00442EB0	. 33C0	xor eax, eax
00442EB2	. 55	push ebp
00442EB3	. 68 322E4400	push <aLoNg3x Loc 442E32>
ebp=0019F400		
00444000	32 13 8B C0	02 00 8B C0
00444010	BC 08 1A 02	D4 20 40 00
00444020	52 75 6E 74	69 6D 65 20

CrackMe by - aLoNg3x - v. 1.00

Name:

Codice:

About - Help

Ok Cancella

地址	数值
0019F378	004240
0019F37C	021A11
0019F380	0041E5

地址	HEX	数据	反汇编	注释
00442EBE	.	8D55 FC	lea edx, [local.1]	
00442EC1	.	8B83 E0020000	mov eax, dword ptr ds:[ebx+0x2E0]	eax=Edit
00442EC7	.	E8 F403FEFF	call <aLoNg3x_.TControl::GetText(void)>	获取密码
00442ECC	.	8B45 FC	mov eax, [local.1]	
00442ECF	.	E8 9C47FCFF	call <aLoNg3x_.Sysutils::StrToInt(System::Ans	将密码转为十进制数字
00442ED4	.	50	push eax	eax=密码
00442ED5	.	8D55 FC	lea edx, [local.1]	
00442ED8	.	8B83 DC020000	mov eax, dword ptr ds:[ebx+0x2DC]	
00442EDE	.	E8 DD03FEFF	call <aLoNg3x_.TControl::GetText(void)>	获取用户名
00442EE3	.	8B45 FC	mov eax, [local.1]	eax=用户名
00442EE6	.	5A	pop edx	
00442EE7	.	E8 08FCFFFF	call <aLoNg3x_.sub_442AF4>	关键函数
00442EEC	.	84C0	test al, al	
00442EEE	.	74 1C	je short <aLoNg3x_.loc_442F0C>	不跳转 按钮消失
00442EF0	.	33D2	xor edx, edx	
00442EF2	.	8B83 D0020000	mov eax, dword ptr ds:[ebx+0x2D0]	
00442EF8	.	E8 B302FEFF	call <aLoNg3x_.Controls::TControl::SetVisible	
00442EFD	.	B2 01	mov dl, 0x1	
004232C0	=	<aLoNg3x_.TControl::GetText(void)>		

首先会获取输入的密码，然后将密码转为十进制数。接着获取用户名，然后是一个关键函数，这个函数如果返回值为1，那么就不跳转，按钮就会消失。接下来分析下这个关键函数。

分析关键函数

IDA View-A	Pseudocode-A	List of applied library modules	Hex View-1	Structures	Enums	Imports
20	v8 = __readfsdword(0);					
21	__writefsdword(0, (unsigned int)&v8);					
22	if (__linkproc__ LStrLen(username_1) <= 5) // 获取用户名长度 比较是否小于等于5					
23	{					
24	bRet = 0;					
25	}					
26	else					
27	{					
28	v2 = sub_442A20(*(unsigned __int8 *)(username_1 + 4) % 7u + 2); // 求阶乘 (username[4]%7)+2					
29	result = 0;					
30	userlen = __linkproc__ LStrLen(username_1); // 求用户名长度					
31	if (userlen > 0) // 如果用户名长度大于0					
32	{					
33	v5 = 1;					
34	do					
35	{					
36	result += v2 * *(unsigned __int8 *)(username_1 + v5++ - 1); // 求出每个用户名的ASCII乘以阶乘的和					
37	--userlen;					
38	}					
39	while (userlen);					
40	}					
41	bRet = result - password;					
42	if (bRet == 31337) // 将结果减去password					
43	LOBYTE(bRet) = 1; // 相等返回1					
44	else					
45	bRet = 0; // 不相等返回0					
46	}					
47	__writefsdword(0, v8);					

这里直接贴出IDA的分析结果，如果想要详细的过程可以去看我的udd文件。首先对用户名的做取模运算之后求阶乘，然后将每个用户名的ASCII乘以阶乘相加，最后把结果减去输入的密码，如果等于31337就返回1，否则返回零。也就是说，我们只要复制上面的过程，然后将结果减去31337，就能得到正确的密钥。

写出注册机

```
//计算第二个按钮Cancel1a所需要的密码
int calckey1()
{
    int key = 0;
```

```
int nTemp = 0;
int num = 0;
char username[20] = { 0 };
printf("请输入用户名,长度必须在六位以上:");
scanf_s("%s", username, 20);

//检查长度
if (strlen(username)<=5)
{
    printf("用户名长度不满足 请重新输入");
    return 0;
}

//求阶乘
num = ((int)username[4]) % 7 + 2;

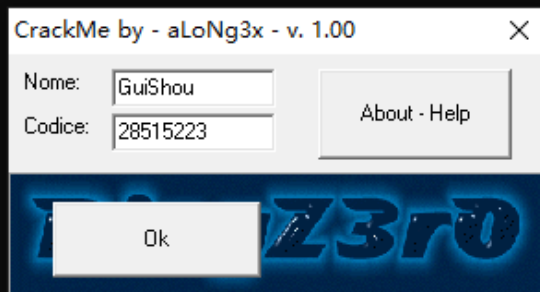
nTemp = num;
for (int i=1;i< nTemp;i++)
{
    num *= i;
}
nTemp = num;
int result = 0;
//求用户名的ASCII和乘以阶乘
for (int i=0;i<strlen(username);i++)
{
    result += nTemp * username[i];
}
//取出正确的密码
key = result - 0x7A69;
printf("%d\n", key);

}
```

验证结果

E:\Projects\Keygen\Debug\Keygen.exe

请输入用户名,长度必须在六位以上:GuiShou
28515223
请按任意键继续. . .



输入计算的结果, 点击 我们可以看到右边的按钮消失了, 并且左边的OK按钮也解除了禁用状态。

分析ok按钮点击事件

接下来来到的位置, 来分析OK按钮的点击事件。

地址	HEX 数据	反汇编	注释
00442D9B	. 8B83 E0020000	mov eax,dword ptr ds:[ebx+0x2E0]	
00442DA1	. E8 1A05FEFF	call <aLoNg3x_.TControl::GetText(void)>	获取密码
00442DA6	. 8B45 FC	mov eax,[local.1]	
00442DA9	. E8 C248FCFF	call <aLoNg3x_.Sysutils::StrToInt(System::A	字符串转数字
00442DAE	. 50	push eax	
00442DAF	. 8D55 FC	lea edx,[local.1]	
00442DB2	. 8B83 DC020000	mov eax,dword ptr ds:[ebx+0x2DC]	
00442DB8	. E8 0305FEFF	call <aLoNg3x_.TControl::GetText(void)>	获取用户名
00442DBD	. 8B45 FC	mov eax,[local.1]	
00442DC0	. 5A	pop edx	
00442DC1	. E8 DAFDFFFF	call <aLoNg3x_.sub_442BA0>	关键函数
00442DC6	. 84C0	test al,al	
00442DC8	. 74 0D	je short <aLoNg3x_.loc_442DD7>	关键跳转
00442DCA	. 33D2	xor edx,edx	
00442DCC	. 8B83 CC020000	mov eax,dword ptr ds:[ebx+0x2CC]	
00442DD2	. E8 D903FEFF	call <aLoNg3x_.Controls::TControl::SetVisib	
00442DD7	> 33C0	xor eax,eax	loc_442DD7
00442DDA	. 5A	pop edx	

还是同样的套路, 有一个关键的算法函数, 这个函数返回1, 按钮就消失, 否则不成功。

分析关键函数

地址	HEX 数据	反汇编	注释
00442BF9	> 8D45 F4	lea eax, [local.3]	loc_442BF9
00442BFC	. E8 0310FCFF	call <_alloca_3>.System::UniqueString(System	
00442C01	. 8D4430 FF	lea eax, dword ptr ds:[eax+esi-0x1]	取密码的最后一位
00442C05	. 50	push eax	
00442C06	. 8B45 F8	mov eax, [local.2]	
00442C09	. 0FB64430 FF	movzx eax, byte ptr ds:[eax+esi-0x1]	eax=pass[i]
00442C0E	. F7E8	imul eax	eax=eax*eax
00442C10	. 0FBFC0	movsx eax, ax	取低4位
00442C13	. F7EE	imul esi	eax=eax*eax*passlen
00442C15	. B9 19000000	mov ecx, 0x19	ecx=0x19
00442C1A	. 99	cdq	
00442C1B	. F7F9	idiv ecx	
00442C1D	. 83C2 41	add edx, 0x41	(eax%0x19)+0x41
00442C20	. 58	pop eax	0019F360
00442C21	. 8810	mov byte ptr ds:[eax], dl	保存结果
00442C23	. 4E	dec esi	长度-1
00442C24	. 85F6	test esi, esi	
00442C26	. ^ 75 D1	jnz short <_alloca_3>.loc_442BE9>	

整个关键函数校验如下，首先取出密码的最后一位，然后做平方运算，接着乘以当前的密码长度，也就是index，然后对0x19取模，最后加上0x41然后保存结果，每个结果对应一位用户名。

写出注册机

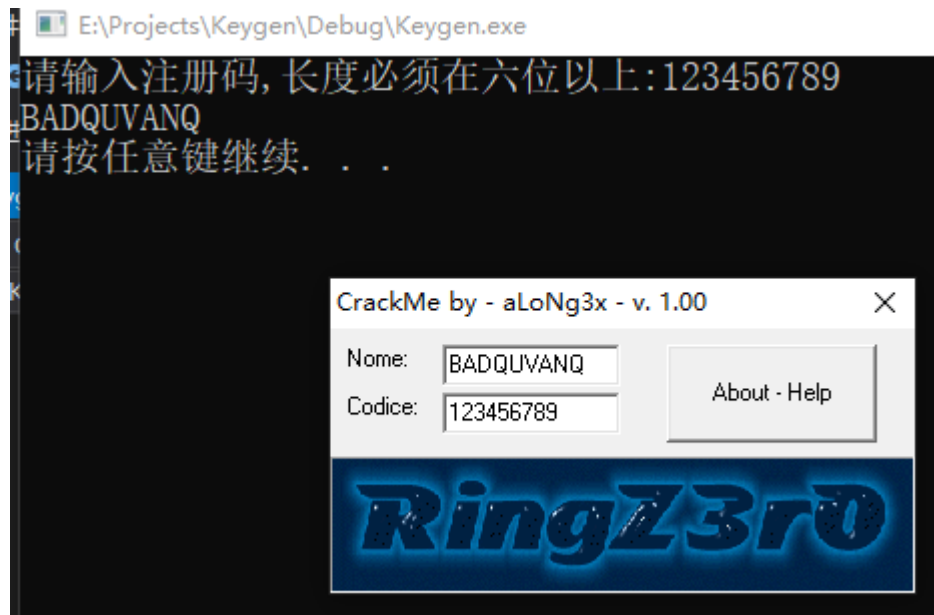
接下来还原算法写出注册机

```
//计算第一个按钮ok所需要的密码
int CalcKey2()
{
    char key[20] = { 0 };
    char username[20] = { 0 };
    printf("请输入注册码,长度必须在六位以上:");

    //输入密码
    scanf_s("%s", key, 20);

    //检查长度
    if (strlen(key) <= 5)
    {
        printf("密码长度不满足 请重新输入");
        return 0;
    }
    for (int i = strlen(key)-1; i!==-1; i--)
    {
        username[i] = (key[i] * key[i] * (i + 1)) % 0x19 + 0x41;
    }
    printf("%s", username);
}
```

校验结果



输入序列号，然后自动生成用户名，可以看到OK按钮也跟着消失了。

需要相关文件的可以到我的Github下载：<https://github.com/TonyChen56/160-Crackme>