Subject: [Script] Hooks.dll
Posted by jnz on Sun, 01 Feb 2009 17:09:50 GMT
View Forum Message <> Reply to Message

Please follow instalation instructions carefuly.

I've created a simple dll that contains some of the hooks that I've used.

Simply create 2 new files in your project and add the following code.

"Hooks.h"
Toggle Spoiler


```
typedef void (*_SerialHook)(int, const char *);
typedef void (*_LoadingEHook)(int, bool);
typedef bool (*_DamageHook)(int, int, int, float, unsigned int);
typedef bool (*_ChatEHook)(int, int, WideStringClass &, int);
typedef void (*_PingHook)(int, int);
typedef bool (*_SuicideHook)(int);
typedef bool (*_RadioHook)(int, int, int, int, int);

typedef void (*_AddSerialHook)(_SerialHook);
typedef void (*_AddLoadingEHook)(_LoadingEHook);
typedef void (*_AddDamageHook)(_DamageHook);
typedef void (*_AddChatEHook)(_ChatEHook);
typedef void (*_AddPingHook)(_PingHook);
typedef void (*_AddSuicideHook)(_SuicideHook);
typedef void (*_AddRadioHook)(_RadioHook);

typedef void (*_RequestSerial)(int, StringClass &);

extern _RequestSerial RequestSerial;

extern _AddSerialHook AddSerialHook;
extern _AddLoadingEHook AddLoadingEHook;
extern _AddDamageHook AddDamageHook;
extern _AddChatEHook AddChatEHook;
extern _AddPingHook AddPingHook;
extern _AddSuicideHook AddSuicideHook;
extern _AddRadioHook AddRadioHook;

void Load_Hooks();
```

"Hooks.cpp"
Toggle Spoiler

```cpp
#include "Windows.h"
#include "stdio.h"
#include "scripts.h"
#include "engine.h"
#include "Hooks.h"

_RequestSerial RequestSerial = 0;
_AddSerialHook AddSerialHook = 0;
_AddLoadingEHook AddLoadingEHook = 0;
_AddDamageHook AddDamageHook = 0;
_AddChatEHook AddChatEHook = 0;
_AddPingHook AddPingHook = 0;
_AddSuicideHook AddSuicideHook = 0;
_AddRadioHook AddRadioHook = 0;

inline void LoadHook(HMODULE dll, void **hook, const char *name)
{
 *hook = (void *)GetProcAddress(dll, name);
 if(!*hook)
 {
  printf("Error loading \"%s\"", name);
  *hook = 0;
 }
}

void Load_Hooks()
{
 HMODULE hooks = LoadLibrary("Hooks.dll");
 LoadHook(hooks, (void **)&RequestSerial, "RequestSerial");
 LoadHook(hooks, (void **)&AddSerialHook, "AddSerialHook");
 LoadHook(hooks, (void **)&AddLoadingEHook, "AddLoadingEHook");
 LoadHook(hooks, (void **)&AddDamageHook, "AddDamageHook");
 LoadHook(hooks, (void **)&AddChatEHook, "AddChatHook");
 LoadHook(hooks, (void **)&AddPingHook, "AddPingHook");
 LoadHook(hooks, (void **)&AddSuicideHook, "AddSuicideHook");
 LoadHook(hooks, (void **)&AddRadioHook, "AddRadioHook");
}
```

Make sure you call

Load_Hooks();

Make sure you load the file attached into the server before you call Load_Hooks.
If you use SSGM, just load it as a plugin in the 01 slot.

Here is a sample SSGM plugin "plugin.cpp" file that uses this:

Toggle Spoiler

```
/* Renegade Scripts.dll
 Example Plugin Code
 Copyright 2007 Whitedragon(MDB), Jonathan Wilson

 This file is part of the Renegade scripts.dll
 The Renegade scripts.dll is free software; you can redistribute it and/or modify it under
 the terms of the GNU General Public License as published by the Free
 Software Foundation; either version 2, or (at your option) any later
 version. See the file COPYING for more details.
 In addition, an exemption is given to allow Run Time Dynamic Linking of this code with any
closed source module that does not contain code covered by this licence.
 Only the source code to the module(s) containing the licenced code has to be released.
*/

/*  This is designed to serve as both an example on how to make a plugin and to give users the
basic framework of a plugin.
    The plugin is simple: it creates an object creation hook and attaches the script
"Plugin_Example_Script" to all objects.
 The script prints out a message whenever an object is created or destroyed.
 There are also examples of the new format for bhs.dll hooks.
*/

#include "scripts.h"
#include <windows.h>
#include "engine.h"
#include "gmmain.h"
#include "plugin.h"
#include "Hooks.h"



void Serial_Hook(int ID, const char *Serial)
{
```

```cpp
 printf("[Serial] %d - %s\n", ID, Serial);
}

void Loading_Hook(int PlayerID, bool IsInGame)
{
 printf("[Load] %d %s\n", PlayerID, IsInGame ? "True" : "False");
}

bool Damage_Hook(int PlayerID, int Damager, int Target, float Damage, unsigned int Warhead)
{
 printf("[Damage] %d %d %d %f %u\n", PlayerID, Damager, Target, Damage, Warhead);
 return 1;
}

bool Chat_Hook(int PlayerID, int Type, WideStringClass &Message, int Target)
{
 printf("[Chat] %d %d %S %d\n", PlayerID, Type, (const wchar_t *)Message, Target);
 return 1;
}

void Ping_Hook(int PlayerID, int PingID)
{
 printf("[Ping] %d %d\n", PlayerID, PingID);
}

bool Suicide_Hook(int PlayerID)
{
 printf("[Suicide] %d\n", PlayerID);
 return 1;
}

bool Radio_Hook(int Team, int PlayerID, int a, int RadioID, int b)
{
 printf("[Radio] %d %d %d %d %d\n", Team, PlayerID, a, RadioID, b);
 return 1;
}

void Plugin_Load()
{
 Load_Hooks();

 AddSerialHook(Serial_Hook);
 AddLoadingEHook(Loading_Hook);
 AddDamageHook(Damage_Hook);
 AddChatEHook(Chat_Hook);
 AddPingHook(Ping_Hook);
 AddSuicideHook(Suicide_Hook);
 AddRadioHook(Radio_Hook);
```

```
}

void Plugin_Unload()
{

}

extern "C" {
DLLEXPORT void SSGM_Player_Join_Hook(int ID, const char *Nick)
{
 StringClass tmp;
 RequestSerial(ID, tmp);
}
}
```

Some of the hook functions allow you to return a bool. For example, the chat hook. If you return 0 you BLOCK the message. With the chat hook, you can also change the message.

```
bool Chat_Hook(int PlayerID, int Type, WideStringClass &Message, int Target)
{
 Message.Format("Hello world!"); //now everyone will always say "Hello world!"
 return 1;
}
```

All of these hooks are compatable with RR, scripts and BIATCH as far as I know. If not, give me a shout.

Updated download.

File Attachments
1) Hooks.dll, downloaded 373 times