
Subject: Need help please

Posted by [Distrbd21](#) on Sat, 26 May 2012 07:01:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

I've been editing Reborns old vet system and trying to learn from it at the same time, It compiles fine but it now crashes my server when it's on.

What I'm trying to get it to do is like MW2 does when you start you have 0 rank points you play and rank up till you get to GA then you can reset it and do it all over again.

So I'm trying to make it like that in ren so you keep your vet points instead of it resetting every map, or when you leave and rejoin, So it's like a Rank system really cause all the players will be ranking up and getting weapons and health and armor, and instead of loosing it when the next map comes or when you leave and rejoin, it will stay so if you play one day and get to 2 1st class, and then you don't play for 2 days, and you get back on, you will still be 2 1st class, and still have all your weapons and health and armor upgrades. ;p

Here's what it say's when it Crashes.

Problem signature:

Problem Event Name: BEX

Application Name: server.dat

Application Version: 1.37.0.1

Application Timestamp: 3e303b7e

Fault Module Name: veteran.dll

Fault Module Version: 0.0.0.0

Fault Module Timestamp: 4fc0afef

Exception Offset: 0000cf3a

Exception Code: c0000417

Exception Data: 00000000

OS Version: 6.1.7601.2.1.0.768.3

Locale ID: 1033

Additional Information 1: 0b66

Additional Information 2: 0b66254370c2d9f2985b0c5de90f53d8

Additional Information 3: 9c99

Additional Information 4: 9c99111c9b5a0ff900d1a25593cdf98e

Code /* 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.
*/
#pragma once
#include "scripts.h"
#include <stdarg.h>
#ifdef WIN32
#define WIN32_LEAN_AND_MEAN
#include <windows.h>
#endif
#include "engine.h"
#include "gmmain.h"
#include "veteran.h"
#include "INI.h"
int ObjectHookID = 0;
ObjectCreateHookStruct *ObjectHookStruct = 0;

void ObjectHookCall(void *data,GameObject *obj) {
// Hooking beacons needs this code, Roshambo kindly provided it.
if(stricmp(Commands->Get_Preset_Name(obj), "Invisible_Object") == 0){
Vector3 pos = Commands->Get_Position(obj);
for(GenericSLNode *x = BaseGameObjList->HeadNode; x != 0; x = x->NodeNext){
GameObject *obj2 = (GameObject *)x->NodeData;
if(obj2 && As_BeaconGameObj(obj2) != 0){
Vector3 objp = Commands->Get_Position(obj2);
if((pos.X == objp.X) && (pos.Y == objp.Y)){
obj = obj2;
break;
}
}
}
}
}

// Is a player
if(Commands->Is_A_Star(obj)){
Attach_Script_Once(obj,"Dis_player","");
}
// Is a vehicle
else if (Is_Vehicle(obj)) {
Attach_Script_Once(obj,"Dis_vehicle","");
}
}

```

```

// Is a beacon
else if (Is_Beacon(obj)) {
Attach_Script_Once(obj,"Dis_beacon","");
}
// Is a c4
else if (Is_C4(obj)) {
Attach_Script_Once(obj,"Dis_c4","");
}
}

```

```

void Plugin_Load() {
ObjectHookStruct = new ObjectCreateHookStruct;
ObjectHookStruct->hook = ObjectHookCall;
ObjectHookStruct->data = 0;
ObjectHookID = AddObjectCreateHook(ObjectHookStruct);
printf("Loaded Remade Vet system by Distrbd21 Source from Reborn\n");
}

```

```

int GetPoints(const char* Preset) {
int VetWorth = 0;
if (strcmp(Preset,"CnC_GDI_Engineer_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_GDI_Engineer_2SF") == 0) VetWorth = 3;
else if (strcmp(Preset,"CnC_GDI_Grenadier_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_GDI_Grenadier_2SF") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_Ignatio_Mobius")) VetWorth = 3;
else if (strcmp(Preset,"CnC_GDI_MiniGunner_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_GDI_MiniGunner_1Off") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_GDI_MiniGunner_2SF") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_GDI_MiniGunner_3Boss")) VetWorth = 3;
else if (strcmp(Preset,"CnC_GDI_RocketSoldier_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_GDI_RocketSoldier_1Off") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_GDI_RocketSoldier_2SF") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_Nod_Engineer_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_Nod_FlameThrower_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_Nod_FlameThrower_1Off") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_Nod_FlameThrower_2SF") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_FlameThrower_3Boss")) VetWorth = 3;
else if (strcmp(Preset,"CnC_Nod_Minigunner_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_Nod_Minigunner_1Off") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_Nod_Minigunner_2SF") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_Minigunner_3Boss")) VetWorth = 3;
else if (strstr(Preset,"CnC_Nod_MiniGunner_3Boss")) VetWorth = 3;
else if (strcmp(Preset,"CnC_Nod_RocketSoldier_0") == 0) VetWorth = 1;
else if (strcmp(Preset,"CnC_Nod_RocketSoldier_1Off") == 0) VetWorth = 2;
else if (strcmp(Preset,"CnC_Nod_RocketSoldier_2SF") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_RocketSoldier_3Boss")) VetWorth = 3;
}

```

```

else if (stricmp(Preset,"CnC_Nod_Technician_0") == 0) VetWorth = 3;
else if (strstr(Preset,"CnC_Sydney_PowerSuit")) VetWorth = 3;
else if (stricmp(Preset,"CnC_Sydney") == 0) VetWorth = 2;
else if (strstr(Preset,"CnC_GDI_APC")) VetWorth = 3;
else if (strstr(Preset,"CnC_GDI_Humm-vee")) VetWorth = 2;
else if (strstr(Preset,"CnC_GDI_MRLS")) VetWorth = 3;
else if (strstr(Preset,"CnC_Civilian_Pickup01_Secret")) VetWorth = 2;
else if (strstr(Preset,"CnC_Civilian_Sedan01_Secret")) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_APC")) VetWorth = 3;
else if (strstr(Preset,"CnC_Nod_Buggy")) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_Mobile Artillery")) VetWorth = 3;
else if (strstr(Preset,"Nod_Chameleon")) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_Recon_Bike")) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_Transport")) VetWorth = 2;
else if (strstr(Preset,"CnC_GDI_Transport")) VetWorth = 2;
else if (strstr(Preset,"CnC_GDI_Medium_Tank")) VetWorth = 4;
else if (strstr(Preset,"CnC_GDI_Mammoth_Tank")) VetWorth = 5;
else if (strstr(Preset,"CnC_Nod_Light_Tank")) VetWorth = 4;
else if (strstr(Preset,"CnC_Nod_Flame_Tank")) VetWorth = 4;
else if (strstr(Preset,"CnC_Nod_Stealth_Tank")) VetWorth = 5;
else if (strstr(Preset,"CnC_Nod_Apache")) VetWorth = 4;
else if (strstr(Preset,"CnC_GDI_Orca")) VetWorth = 4;
else if (strstr(Preset,"CnC_GDI_Harvester")) VetWorth = 2;
else if (strstr(Preset,"CnC_Nod_Harvester")) VetWorth = 2;
return VetWorth;
}

```

```

struct VeteranPlayers {
    std::string PlayerName;
    int VeteranPoints;
};

```

```
std::vector<VeteranPlayers> VetInfo;
```

```

struct RepPlayers {
    std::string RepPlayerName;
    int RepPoints;
};

```

```
std::vector<RepPlayers> RepInfo;
```

```

void VetUpdate(int ID, int number, int score) {

    if (!VetInfo.empty()) {

```

```

for (int i = 0; i < VetInfo.size(); i++) {
if (VetInfo[i].PlayerName == Get_Player_Name_By_ID(ID)) {
int cvp, now;
cvp = VetInfo[i].VeteranPoints;
now = number + cvp;
VetInfo[i].VeteranPoints = now;

int Team;
Team = Get_Team(ID);
if (cvp < 540 && now >= 540) {
char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General of the Army",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
if (Team == 0) {
sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General of the Army, enjoy the extra 28HP armor.", ID);
Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
sprintf(Promote, "ppage %d Locke himself has promoted you to a General of the Army, enjoy
the extra 28HP armor.", ID);
Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 520 && now >= 520) {
char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
if (Team == 0) {
sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 500 && now >= 500) {
char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
}
}
}

```

```

if (Team == 0) {
    sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 480 && now >= 480) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 460 && now >= 460) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 440 && now >= 440) {
    char Promote[255];

```

```

    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 420 && now >= 420) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 400 && now >= 400) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}

```

```

    }
    if (cvp < 380 && now >= 380) {
        char Promote[255];
        sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
        Console_Input(Promote);
        if (Team == 0) {
            sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
            Commands->Create_2D_WAV_Sound("bonus_complete.wav");
        }
        else {
            sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
            Commands->Create_2D_WAV_Sound("bonus_complete.wav");
        }
        Console_Input(Promote);
    }
    if (cvp < 360 && now >= 360) {
        char Promote[255];
        sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
        Console_Input(Promote);
        if (Team == 0) {
            sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
            Commands->Create_2D_WAV_Sound("bonus_complete.wav");
        }
        else {
            sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
            Commands->Create_2D_WAV_Sound("bonus_complete.wav");
        }
        Console_Input(Promote);
    }
    if (cvp < 340 && now >= 340) {
        char Promote[255];
        sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
        Console_Input(Promote);
        if (Team == 0) {
            sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
            Commands->Create_2D_WAV_Sound("bonus_complete.wav");
        }
        else {
            sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);

```



```

    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 320 && now >= 320) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 300 && now >= 300) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 280 && now >= 280) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
}

```

```

else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 260 && now >= 260) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 240 && now >= 240) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 220 && now >= 220) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to

```

```

General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 200 && now >= 200) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 180 && now >= 180) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 160 && now >= 160) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));

```

```

Console_Input(Promote);
if (Team == 0) {
    sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 140 && now >= 140) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 120 && now >= 120) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
}
if (cvp < 100 && now >= 100) {

```

```

char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
if (Team == 0) {
    sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 80 && now >= 80) {
char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
if (Team == 0) {
    sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
Console_Input(Promote);
}
if (cvp < 60 && now >= 60) {
char Promote[255];
sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
Console_Input(Promote);
if (Team == 0) {
    sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
else {
    sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
    Commands->Create_2D_WAV_Sound("bonus_complete.wav");
}
}

```

```

Console_Input(Promote);
    }
if (cvp < 50 && now >= 50) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
    }
if (cvp < 40 && now >= 40) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of General",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane has recognised your vision, you have been promoted to
General, enjoy the extra 45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    else {
        sprintf(Promote, "ppage %d Locke himself has promoted you to a General, enjoy the extra
45HP armor.", ID);
        Commands->Create_2D_WAV_Sound("bonus_complete.wav");
    }
    Console_Input(Promote);
    }
else if (cvp < 20 && now >= 20) {
    char Promote[255];
    sprintf(Promote,"msg Veteran: %s just got promoted to the rank of Brigadier",
Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d The Brotherhood of Nod likes your style, you are promoted to
rank Brigadier, enjoy the extra 30HP armor.", ID);
        Commands->Create_2D_WAV_Sound("m00bgwf_kill0031i1nbrs_snd.wav");
    }
    else {
        sprintf(Promote, "ppage %d Keep em coming! you done all that left handed? You are

```

```
promoted to Brigadier, enjoy the extra 30HP armor.", ID);
    Commands->Create_2D_WAV_Sound("m00achk_kill0034i1gbmg_snd.wav");
}
Console_Input(Promote);
}
else if (cvp < 0 && now >= 0) {
    char Promote[255];
    sprintf(Promote, "msg Veteran: %s just got promoted to the rank of Major, he gets an extra
15HP armor on every character he now buys", Get_Player_Name_By_ID(ID));
    Console_Input(Promote);
    if (Team == 0) {
        sprintf(Promote, "ppage %d Kane wants you in his elite, you are promoted to Major, enjoy the
splendors of war brother with an extra 15HP armor", ID);
        Commands->Create_2D_WAV_Sound("m00avis_kick0041i1nsmg_snd.wav");
    }
    else {
        sprintf(Promote, "ppage %d The deadsix needs people like you, you are promoted to Major!
He will have an extra 15HP armor", ID);
        Commands->Create_2D_WAV_Sound("m00_wins0003eval_snd.wav");
    }
    Console_Input(Promote);
}
break;
}
}
}
}
```

```
void RepUpdate(int ID, int number) {

if (!RepInfo.empty()) {
for (int i = 0; i < RepInfo.size(); i++) {
if (RepInfo[i].RepPlayerName == Get_Player_Name_By_ID(ID)) {
int cvp, now;
cvp = RepInfo[i].RepPoints;
now = number + cvp;
RepInfo[i].RepPoints = now;

if (cvp < 200 && now >= 200) {
VetUpdate(ID, 1, Commands->Get_Points(Get_GameObj(ID)));
Display_Int_Player(Get_GameObj(ID),1,"Bonus repairing veteran points gained: 1 ");
RepInfo[i].RepPoints = 0;
now = 0;
cvp = 0;
}
}
```

```
    break;
}
}
}
```

```
int VetCheckPoints(int ID) {
if (!VetInfo.empty()) {
for (int i = 0; i < VetInfo.size(); i++) {
if (VetInfo[i].PlayerName == Get_Player_Name_By_ID(ID)) {
int Points;
Points = VetInfo[i].VeteranPoints;
return Points;
}
}
}
return 0;
}
```

```
int RepCheckPoints(int ID) {
if (!ReplInfo.empty()) {
for (int i = 0; i < ReplInfo.size(); i++) {
if (ReplInfo[i].RepPlayerName == Get_Player_Name_By_ID(ID)) {
int Points;
Points = ReplInfo[i].RepPoints;
return Points;
}
}
}
return 0;
}
```

```
bool VetCheck(int ID) {
if (!VetInfo.empty()) {
for (int i = 0; i < VetInfo.size(); i++) {
if (VetInfo[i].PlayerName == Get_Player_Name_By_ID(ID)) {
return true;
}
}
}
return false;
}
```

```
bool RepCheck(int ID) {
if (!ReplInfo.empty()) {
```



```

for (int i = 0; i < RepInfo.size(); i++) {
    if (RepInfo[i].RepPlayerName == Get_Player_Name_By_ID(ID)) {
        return true;
    }
}
return false;
}

void VetAddPlayer(int ID) {
    if (VetCheck(ID) == false) {
        VeteranPlayers temp;
        temp.PlayerName = Get_Player_Name_By_ID(ID);
        temp.VeteranPoints = 0;
        VetInfo.push_back(temp);
    }
}

void RepAddPlayer(int ID) {
    if (RepCheck(ID) == false) {
        RepPlayers temp;
        temp.RepPlayerName = Get_Player_Name_By_ID(ID);
        temp.RepPoints = 0;
        RepInfo.push_back(temp);
    }
}

void VetClearPlayers() {
    VetInfo.erase(VetInfo.begin(), VetInfo.end());
    int Players = The_Game()->MaxPlayers;
    for (int i = 1; i <= Players; i++) {
        if (Get_GameObj(i)) {
            VetAddPlayer(i);
        }
    }
}

void RepClearPlayers() {
    RepInfo.erase(RepInfo.begin(), RepInfo.end());
    int Players = The_Game()->MaxPlayers;
    for (int i = 1; i <= Players; i++) {
        if (Get_GameObj(i)) {
            RepAddPlayer(i);
        }
    }
}

```

```

void Dis_vet_System::Created(GameObject *obj) {
int Points;
Points = VetCheckPoints(Get_Player_ID(obj));
if (Points >= 540) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 28));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 520) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 27));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 500) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 26));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 480) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 25));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 460) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 24));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 440) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 23));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 420) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 22));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 400) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 21));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
}

```

```

if (Points >= 380) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 20));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 360) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 19));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 340) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 18));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 320) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 17));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 300) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 16));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 280) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 15));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 260) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 14));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 240) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 13));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 220) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 12));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 200) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 11));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
}

```

```

Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 180) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 10));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 160) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 9));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 140) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 8));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 120) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 7));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 100) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 6));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 80) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 5));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 60) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 4));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 40) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 3));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 20) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 2));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
}
if (Points >= 0) {

```

```
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 0));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
}
```

```
Commands->Start_Timer(obj, this, 0.1f, 1);
}
```

```
void Dis_vet_System::Timer_Expired(GameObject *obj, int number) {
```

```
if (number == 1) {
    int Points;
    Points = VetCheckPoints(Get_Player_ID(obj));
    static bool GA = true;
    static bool GEN = true;
    static bool LTG = true;
    static bool MG = true;
    static bool BG = true;
    static bool COL = true;
    static bool LTC = true;
    static bool MAJ = true;
    static bool CPT = true;
    static bool FLT = true;
    static bool SLT = true;
    static bool CW5 = true;
    static bool CW4 = true;
    static bool CW3 = true;
    static bool CW2 = true;
    static bool WO1 = true;
    static bool SMA = true;
    static bool CSM = true;
    static bool SGM = true;
    static bool FSG = true;
    static bool MSG = true;
    static bool SFC = true;
    static bool SSG = true;
    static bool SGT = true;
    static bool CPL = true;
    static bool SPC = true;
    static bool PFC = true;
    static bool PVT = true;

    if (Points >= 540 && GA) {
        Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 28));
    }
}
```

```

Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
GA = false;
}
if (Points >= 520 && Points < 540 && GEN) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 27));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
GEN = false;
}
if (Points >= 500 && Points < 520 && LTG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 26));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
LTG = false;
}
if (Points >= 480 && Points < 500 && MG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 25));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
MG = false;
}
if (Points >= 460 && Points < 480 && BG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 24));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
BG = false;
}
if (Points >= 440 && Points < 460 && COL) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 23));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
COL = false;
}
if (Points >= 420 && Points < 440 && LTC) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 22));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
LTC = false;
}
if (Points >= 400 && Points < 420 && MAJ) {

```

```

Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 21));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
MAJ = false;
}
if (Points >= 380 && Points < 400 && CPT) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 20));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CPT = false;
}
if (Points >= 360 && Points < 380 && FLT) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 19));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
FLT = false;
}
if (Points >= 340 && Points < 360 && SLT) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 18));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
SLT = false;
}
if (Points >= 320 && Points < 340 && CW5) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 17));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CW5 = false;
}
if (Points >= 300 && Points < 320 && CW4) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 16));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CW4 = false;
}
if (Points >= 280 && Points < 300 && CW3) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 15));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CW3 = false;
}
}

```

```

if (Points >= 260 && Points < 280 && CW2) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 14));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CW2 = false;
}
if (Points >= 240 && Points < 260 && WO1) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 13));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
WO1 = false;
}
if (Points >= 220 && Points < 240 && SMA) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 12));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
SMA = false;
}
if (Points >= 200 && Points < 220 && CSM) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 11));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
CSM = false;
}
if (Points >= 180 && Points < 200 && SGM) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 10));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
SGM = false;
}
if (Points >= 160 && Points < 180 && FSG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 9));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Railgun_Player", 1);
FSG = false;
}
if (Points >= 140 && Points < 160 && MSG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 8));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "CnC_POW_RepairGun_Player", 1);
MSG = false;
}

```



```

}
if (Points >= 120 && Points < 140 && SFC) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 7));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_TiberiumFlechetteGun_Player", 1);
SFC = false;
}
if (Points >= 100 && Points < 120 && SSG) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 6));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_SniperRifle_Player", 1);
SSG = false;
}
if (Points >= 80 && Points < 100 && SGT) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 5));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_LaserChaingun_Player", 1);
SGT = false;
}
if (Points >= 60 && Points < 80 && CPL) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 4));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_ChemSprayer_Player", 1);
CPL = false;
}
if (Points >= 40 && Points < 60 && SPC) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 3));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Chaingun_Player", 1);
SPC = false;
}
if (Points >= 20 && Points < 40 && PFC) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 2));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
Commands->Give_Powerup(obj, "POW_Shotgun_Player", 1);
PFC = false;
}
if (Points >= 0 && Points < 20 && PVT) {
Set_Max_Shield_Strength(obj, (Commands->Get_Max_Shield_Strength(obj) + 1));
Commands->Set_Health(obj, Commands->Get_Max_Health(obj));
Commands->Set_Shield_Strength(obj, Commands->Get_Max_Shield_Strength(obj));
PVT = false;
}

```

```

}
Commands->Start_Timer(obj, this, 1.0f, 1);
}
}

```

```

void Dis_vet_System::Killed(GameObject *obj, GameObject *shooter) {

    Destroy_Script();
}

```

```

ScriptRegistrant<Dis_vet_System> Dis_vet_System_Registrant("Dis_vet_System","");

```

```

void Dis_player::Created(GameObject *obj) {
    Commands->Attach_Script(obj,"Dis_vet_System","");
}

```

```

void Dis_player::Damaged(GameObject *obj, GameObject *damager, float damage) {
    if(damage < 0){
        if (Commands->Is_A_Star(damager) && Commands->Get_Player_Type(damager) ==
Commands->Get_Player_Type(obj)){
            RepUpdate(Get_Player_ID(damager), ((damage - damage) - damage));
        }
    }
}

```

```

void Dis_player::Killed(GameObject *obj, GameObject *shooter) {

    if (Commands->Is_A_Star(shooter) && Commands->Get_Player_Type(shooter) !=
Commands->Get_Player_Type(obj)){
        VetUpdate(Get_Player_ID(shooter), GetPoints(Commands->Get_Preset_Name(obj)),
Commands->Get_Points(shooter));
        Display_Int_Player(shooter,GetPoints(Commands->Get_Preset_Name(obj)),"Veteran points
gained: %d "),GetPoints(Commands->Get_Preset_Name(obj));
    }
    else if(Commands->Is_A_Star(shooter) && Get_Player_ID(shooter) == Get_Player_ID(obj)){
        VetUpdate(Get_Player_ID(shooter), (GetPoints(Commands->Get_Preset_Name(obj)) * -1),
Commands->Get_Points(shooter));
        Display_Int_Player(shooter,(GetPoints(Commands->Get_Preset_Name(obj)) * -1),"Veteran points
lost: %d "),GetPoints(Commands->Get_Preset_Name(obj));
    }
}
}

```

```
ScriptRegistrant<Dis_player> Dis_player_Registrant("Dis_player","");
```

```
void Dis_vehicle::Damaged(GameObject *obj, GameObject *damager, float damage) {  
    if(damage < 0){  
        if (Commands->Is_A_Star(damager) && Commands->Get_Player_Type(damager) ==  
Commands->Get_Player_Type(obj)){  
            RepUpdate(Get_Player_ID(damager), ((damage - damage) - damage)/2);  
        }  
    }  
}
```

```
void Dis_vehicle::Killed(GameObject *obj, GameObject *shooter) {  
    if (Commands->Is_A_Star(shooter)){  
        VetUpdate(Get_Player_ID(shooter), GetPoints(Commands->Get_Preset_Name(obj)),  
Commands->Get_Points(shooter));  
        Display_Int_Player(shooter,GetPoints(Commands->Get_Preset_Name(obj)),"Veteran points  
gained: %d "),GetPoints(Commands->Get_Preset_Name(obj));  
    }  
}
```

```
ScriptRegistrant<Dis_vehicle> Dis_vehicle_Registrant("Dis_vehicle","");
```

```
void Dis_building::Damaged(GameObject *obj, GameObject *damager, float damage) {  
    if(damage < 0){  
        if (Commands->Is_A_Star(damager) && Commands->Get_Player_Type(damager) ==  
Get_Object_Type(obj)){  
            RepUpdate(Get_Player_ID(damager), ((damage - damage) - damage));  
        }  
    }  
}
```

```
void Dis_building::Killed(GameObject *obj, GameObject *shooter) {  
    if (Commands->Is_A_Star(shooter)){  
        VetUpdate(Get_Player_ID(shooter), 25, Commands->Get_Points(shooter));  
        Display_Int_Player(shooter,25,"Veteran points gained: 35 ");  
    }  
}
```

```
ScriptRegistrant<Dis_building> Dis_building_Registrant("Dis_building","");
```

```

void Dis_beacon::Killed(GameObject *obj, GameObject *shooter) {
    VetUpdate(Get_Player_ID(shooter), 5, Commands->Get_Points(shooter));
    Display_Int_Player(shooter,5,"bonus veteran points gained: 10 ");
}

```

```

ScriptRegistrant<Dis_beacon> Dis_beacon_Registrant("Dis_beacon","");

```

```

void Dis_c4::Killed(GameObject *obj, GameObject *shooter) {
    VetUpdate(Get_Player_ID(shooter), 1, Commands->Get_Points(shooter));
    Display_Int_Player(shooter,1,"Bonus veteran points gained: 5 ");
}

```

```

ScriptRegistrant<Dis_c4> Dis_c4_Registrant("Dis_c4","");

```

```

class vetChatCommand : public ChatCommandClass {
void Triggered(int ID,const TokenClass &Text,int ChatType) {
    GameObject *obj = Get_GameObj(ID);
    if (!VetInfo.empty()) {
        for (int i = 0; i < VetInfo.size(); i++) {
            if (VetInfo[i].PlayerName == Get_Player_Name_By_ID(ID)) {
                int vetp;
                float score;
                vetp = VetInfo[i].VeteranPoints;
                score = Commands->Get_Points(Get_GameObj(ID));

                char Vet[255];
                if (vetp >= 540) {
                    sprintf(Vet, "page %d Veteran Info for %s - Veteran Rank: General Of The Army", ID,
Get_Player_Name_By_ID(ID));
                }
                else if (vetp >= 520) {
                    sprintf(Vet, "page %d Veteran Info for %s - Veteran Points %d/520 - Veteran Rank:
General", ID, Get_Player_Name_By_ID(ID), vetp);
                }
                else if (vetp >= 500) {
                    sprintf(Vet, "page %d Veteran Info for %s - Veteran Points %d/500 - Veteran Rank:
Lieutenant General", ID, Get_Player_Name_By_ID(ID), vetp);
                }
            }
        }
    }
}
}

```

```

else if (vetp >= 480) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/480 - Veteran Rank: Major
General", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 460) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/460 - Veteran Rank: Brigadier
General", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 440) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/440 - Veteran Rank:
Colonel", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 420) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/420 - Veteran Rank:
Lieutenant Colonel", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 400) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/400 - Veteran Rank: Major",
ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 380) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/380 - Veteran Rank:
Captain", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 360) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/360 - Veteran Rank: First
Lieutenant", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 340) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/340 - Veteran Rank: Second
Lieutenant", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 320) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/320 - Veteran Rank: Chief
Warrant Officer 5", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 300) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/300 - Veteran Rank: Chief
Warrant Officer 4", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 280) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/280 - Veteran Rank: Chief
Warrant Officer 3", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 260) {
    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/260 - Veteran Rank: Chief
Warrant Officer 2", ID, Get_Player_Name_By_ID(ID), vetp);
}
}

```

```

else if (vetp >= 240) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/240 - Veteran Rank: Warrant
Officer", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 220) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/220 - Veteran Rank:
Sergeant Major of the Army", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 200) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/200 - Veteran Rank:
Command Sergeant Major", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 180) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/180 - Veteran Rank:
Sergeant Major", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 160) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/160 - Veteran Rank: First
Sergeant", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 140) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/140 - Veteran Rank: Master
Sergeant", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 120) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/120 - Veteran Rank:
Sergeant First Class", ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 100) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/100 - Veteran Rank: Staff
Sergeant", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 80) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/80 - Veteran Rank:
Sergeant", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 60) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/60 - Veteran Rank: Corporal",
ID, Get_Player_Name_By_ID(ID));
}
else if (vetp >= 40) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/40 - Veteran Rank:
Specialist", ID, Get_Player_Name_By_ID(ID), vetp);
}
else if (vetp >= 20) {
    printf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/20 - Veteran Rank: Private
First Class", ID, Get_Player_Name_By_ID(ID), vetp);
}

```

```

    else {
        sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points 0/%d - Veteran Rank: Private",
ID, Get_Player_Name_By_ID(ID), vetp);

    }
    Console_Input(Vet);
    break;
}
}
};
ChatCommandRegistrant<vetChatCommand>
vetChatCommandReg("!vet;!vetinfo",CHATTYPE_ALL,0,GAMEMODE_AOW);

```

```

void Plugin_Unload() {
printf("Un-loaded Remade Vet system by Distrbd21 Source from Reborn\n");
}

```

```

extern "C" {

const char *RANKS_FILE = "C:\\\\Westwood\\DSGInfo\\ranks.ini";
DLLEXPORT void SSGM_Player_Join_Hook(int ID, const char *Nick) {
    VetAddPlayer(ID);
    RepAddPlayer(ID);
    VetInfo[ID].VeteranPoints = INI::ReadInt("vet", Nick, RANKS_FILE);
    RepInfo[ID].RepPoints = INI::ReadInt("rep", Nick, RANKS_FILE);
}
DLLEXPORT void SSGM_Player_Leave_Hook(int ID) {
    int Kills =Get_Kills(ID) + INI::ReadInt("Kills", Get_Player_Name_By_ID(ID), RANKS_FILE);
    INI::Write("Kills", Get_Player_Name_By_ID(ID), Kills, RANKS_FILE);
    int Deaths =Get_Deaths(ID) + INI::ReadInt("Deaths", Get_Player_Name_By_ID(ID),
RANKS_FILE);
    INI::Write("Deaths", Get_Player_Name_By_ID(ID), Deaths, RANKS_FILE);
    INI::Write("vet", Get_Player_Name_By_ID(ID), VetInfo[ID].VeteranPoints, RANKS_FILE);
    INI::Write("rep", Get_Player_Name_By_ID(ID), RepInfo[ID].RepPoints, RANKS_FILE);
}

DLLEXPORT void SSGM_GameOver_Hook(int ID) {
    int Kills =Get_Kills(ID) + INI::ReadInt("Kills", Get_Player_Name_By_ID(ID), RANKS_FILE);

```

```
INI::Write("Kills", Get_Player_Name_By_ID(ID), Kills, RANKS_FILE);
int Deaths =Get_Deaths(ID) + INI::ReadInt("Deaths", Get_Player_Name_By_ID(ID),
RANKS_FILE);
INI::Write("Deaths", Get_Player_Name_By_ID(ID), Deaths, RANKS_FILE);
//VetClearPlayers();
//RepClearPlayers();
}
```

```
DLLEXPORT void SSGM_Level_Loaded_Hook() {
//Cannot hook buildings as they are not created, but there as part of the map, have to attach vet
code here on map load
```

```
//Doesn't work for some reason :S
//Attach_Script_Building("Dis_building", "", -1);
//Use this instead
GenericSLNode *x = BuildingGameObjList->HeadNode;
while (x){
    GameObject *o = As_BuildingGameObj((GameObject *)x->NodeData);
    if (o && Commands->Get_Health(o) > 0){
Attach_Script_Once(o,"Dis_building","");
    }
    x = x->NodeNext;
}
}

}[/align]
```
