
Subject: Re: Veteran Plugin.

Posted by [Gen_Blacky](#) on Tue, 04 Dec 2007 00:09:55 GMT

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

Hex wrote on Mon, 03 December 2007 14:30Gen_Blacky wrote on Sun, 02 December 2007 14:19im surprised some 1 would release there Veteran system. theres like 3 or more going around in ren but people don't like sharing.

People do share, but some get annoyed when others demand the source be released

yes that is true

reborn wrote on Mon, 03 December 2007 10:55Gen_Blacky wrote on Sun, 02 December 2007 14:19im surprised some 1 would release there Veteran system. theres like 3 or more going around in ren but people don't like sharing.

```
/**
*****
***** VETERAN STUFF *****
**/
```

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

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

```
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 < 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 on all NEW characters", ID);
```



```

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;
}

```

```

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;
}

```

```

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 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 reb_vet_System::Created(GameObject *obj) {
    int Points;
    Points = VetCheckPoints(Get_Player_ID(obj));
    if (Points >= 100) {

```

```

Level = 3;

GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
IconID = Commands->Get_ID(Icon);
Commands->Set_Model(Icon, "p_keycrd_red");
Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
Set_Max_Shield_Strength(obj, 145);
Commands->Set_Shield_Strength(obj, 145);

Commands->Start_Timer(obj, this, 2.0f, 3);
}
else if (Points >= 60) {
Level = 2;

GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
Commands->Set_Model(Icon, "p_keycrd_yel");
Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
IconID = Commands->Get_ID(Icon);

Set_Max_Shield_Strength(obj, 130);
Commands->Set_Shield_Strength(obj, 130);

}
else if (Points >= 30) {

Level = 1;

GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
Commands->Set_Model(Icon, "p_keycrd_grn");
Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
IconID = Commands->Get_ID(Icon);

if (strstr(Commands->Get_Preset_Name(obj),"CnC_Nod_FlameThrower_2SF")){

}

Set_Max_Shield_Strength(obj, 115);
Commands->Set_Shield_Strength(obj, 115);

}
else {
Level = 0;

```

```

}
Commands->Start_Timer(obj, this, 60.0f, 10);
}

void reb_vet_System::Custom(GameObject *obj, int message, int param, GameObject *sender) {
if (message == CUSTOM_EVENT_VEHICLE_ENTER) {
    Commands->Destroy_Object(Commands->Find_Object(IconID));
}
else if (Level == 3) {
    GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
    Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
    Commands->Set_Model(Icon, "p_keycrd_red");
    Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
    IconID = Commands->Get_ID(Icon);
    float Shield;
    Shield = Commands->Get_Shield_Strength(obj);
    Set_Max_Shield_Strength(obj, 145);
    Commands->Set_Shield_Strength(obj, Shield);

}
else if (Level == 2) {
    GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
    Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
    Commands->Set_Model(Icon, "p_keycrd_yel");
    Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
    IconID = Commands->Get_ID(Icon);
    float Shield;
    Shield = Commands->Get_Shield_Strength(obj);
    Set_Max_Shield_Strength(obj, 130);
    Commands->Set_Shield_Strength(obj, Shield);

}
else if (Level == 1) {
    GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
    Commands->Attach_Script(Icon,"MDB_SSGM_Destroy_When_Object_Destroyed",ToString(Commands->Get_ID(obj)).c_str());
    IconID = Commands->Get_ID(Icon);
    Commands->Set_Model(Icon, "p_keycrd_grn");
    Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");

    if (strstr(Commands->Get_Preset_Name(obj),"CnC_Nod_FlameThrower_2SF")){

}

float Shield;
Shield = Commands->Get_Shield_Strength(obj);

```

```

Set_Max_Shield_Strength(obj, 115);
Commands->Set_Shield_Strength(obj, Shield);
}
Commands->Start_Timer(obj, this, 60.0f, 10);
}

```

```

void reb_vet_System::Timer_Expired(GameObject *obj, int number) {
/*
if (number == 3) {
    Commands->Apply_Damage(obj, -1.0f, "Repair", false);
    Commands->Start_Timer(obj, this, 2.0f, 5);
}
else if (number == 10) {
    int Points;
    Points = VetCheckPoints(Get_Player_ID(obj));
    if (Points >= 100 && Level < 3) {
        Level = 3;
        if (Get_Vehicle(obj) == 0) {
            Commands->Destroy_Object(Commands->Find_Object(ID));
            GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
            Commands->Set_Model(Icon, "p_keycrd_red");
            Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
            ID = Commands->Get_ID(Icon);

        }
        Commands->Start_Timer(obj, this, 2.0f, 3);
    }
    else if (Points >= 60 && Level < 2) {
        Level = 2;
        if (Get_Vehicle(obj) == 0) {
            Commands->Destroy_Object(Commands->Find_Object(ID));
            GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
            Commands->Set_Model(Icon, "p_keycrd_yel");
            Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");
            ID = Commands->Get_ID(Icon);

        }
    }
    else if (Points >= 30 && Level < 1) {
        Level = 1;
        if (Get_Vehicle(obj) == 0) {
            Commands->Destroy_Object(Commands->Find_Object(ID));
            GameObject *Icon = Commands->Create_Object_At_Bone(obj,"Invisible_Object", "c L THIGH");
            Commands->Set_Model(Icon, "p_keycrd_grn");
            Commands->Attach_To_Object_Bone(Icon, obj, "c L THIGH");

```

```

ID = Commands->Get_ID(Icon);

}
}
Commands->Start_Timer(obj, this, 60.0f, 10);
}
*/
}

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

    Destroy_Script();
}

void reb_vet_System::Destroyed(GameObject *obj) {
    Commands->Destroy_Object(Commands->Find_Object(IconID));
    Destroy_Script();
}

void reb_vet_System_Vehicles::Custom(GameObject *obj, int message, int param, GameObject
*sender) {
    if (message == CUSTOM_EVENT_VEHICLE_ENTER) {
        int Points;
        Points = VetCheckPoints(Get_Player_ID(sender));
        if (Points >= 100) {
            Level = 3;
            IsPoweredUp = true;

            if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing
            }
            else {
                Commands->Start_Timer(obj, this, 1.0f, 3);
            }
        }
        else if (Points >= 60) {
            Level = 2;
            IsPoweredUp = true;

            if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing

                Commands->Start_Timer(obj, this, 1.0f, 2);
            }
            else {
                Commands->Start_Timer(obj, this, 1.0f, 3);
            }
        }
    }
}

```

```

else if (Points >= 30) {
    Level = 1;
    IsPoweredUp = true;

    if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing
    }
    else {
        Commands->Start_Timer(obj, this, 1.0f, 1);
    }
}
Commands->Send_Custom_Event(sender, sender, CUSTOM_EVENT_VEHICLE_ENTER, 0, 0);
}
else if (message == CUSTOM_EVENT_VEHICLE_EXIT) {
    if (IsPoweredUp == true) {
        IsPoweredUp = false;
        if (Level == 1) {
            if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing
            }
        }
        else if (Level == 2) {
            if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing
            }
        }
        else if (Level == 3) {
            if (IsLowClassVehicle(Commands->Get_Preset_Name(obj)) == true) {
//nothing
            }
        }
    }
    Commands->Send_Custom_Event(sender, sender, CUSTOM_EVENT_VEHICLE_EXIT, 0, 0);
}
}
void reb_vet_System_Vehicles::Timer_Expired(GameObject *obj, int number) {
    if (number == 3 && IsPoweredUp == true) {
        Commands->Destroy_Object(Commands->Find_Object(ID));
        Commands->Apply_Damage(obj, -3.0f, "Repair", false);
        Commands->Start_Timer(obj, this, 1.0f, 3);
    }
    else if (number == 2 && IsPoweredUp == true) {
        float Health, MaxHealth;
        Health = Commands->Get_Health(obj);
        MaxHealth = Commands->Get_Max_Health(obj);
        if (Health == MaxHealth) {
            // Nothing
        }
    }
}

```



```

else if (Health >= MaxHealth - 2) {
    Commands->Set_Health(obj, MaxHealth);
}
else if (Health <= MaxHealth - 3) {
    Commands->Set_Health(obj, Health + 3);
}
Commands->Start_Timer(obj, this, 1.0f, 2);
}
else if (number == 1 && IsPoweredUp == true) {
    float Shield, MaxShield;
    Shield = Commands->Get_Shield_Strength(obj);
    MaxShield = Commands->Get_Max_Shield_Strength(obj);
    if (Shield == MaxShield) {
        // Nothing
    }
    else if (Shield >= MaxShield - 2) {
        Commands->Set_Shield_Strength(obj, MaxShield);
    }
    else if (Shield <= MaxShield - 3) {
        Commands->Set_Shield_Strength(obj, Shield + 3);
    }
    Commands->Start_Timer(obj, this, 1.0f, 1);
}
}

void reb_vet_System_Vehicles::Killed(GameObject *obj, GameObject *shooter) {
    Commands->Destroy_Object(Commands->Find_Object(ID));
    Destroy_Script();
}

void reb_vet_System_Vehicles::Destroyed(GameObject *obj) {
    Commands->Destroy_Object(Commands->Find_Object(ID));
    Destroy_Script();
}

void Player_Join_Hook(int i,const char *Nick) {
    VetAddPlayer(i);
    if (!Data->Plugins.empty()) {
        std::vector<PluginInfo*>::const_iterator it;
        for (it = Data->Plugins.begin();it != Data->Plugins.end(); ++it) {
            if ((*it)->Type == Plugin) {
                if ((*it)->PlayerJoinHookHandle) {
                    (*it)->PlayerJoinHookHandle(i,Nick);
                }
            }
        }
    }
}

```

```

}
if (Settings->GameMode == 2) {
    CTF_Player_Join(i);
}
}
}

```

```

void GameOver() {
    VetClearPlayers();
    char ObjectsType[10],ObjectsType2[10],ObjectsFile[20];
    // rest of game over code from ssgm here, just didnt paste it to spam
}

```

```

ScriptRegistrant<reb_vet_System> reb_vet_System_Registrant("reb_vet_System","");
ScriptRegistrant<reb_vet_System_Vehicles>
reb_vet_System_Vehicles_Registrant("reb_vet_System_Vehicles","");

```

```

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 >= 100) {
                    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Rank: General", ID,
Get_Player_Name_By_ID(ID));
                }
                else if (vetp >= 60) {
                    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/100 - Veteran Rank:
Brigadier", ID, Get_Player_Name_By_ID(ID), vetp);
                }
                else if (vetp >= 30) {
                    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/60 - Veteran Rank: Major",
ID, Get_Player_Name_By_ID(ID), vetp);
                }
                else {

```

```

    sprintf(Vet, "ppage %d Veteran Info for %s - Veteran Points %d/30 - Veteran Rank: Private",
ID, Get_Player_Name_By_ID(ID), vetp);
    }
    Console_Input(Vet);
    break;
    }
}
};
ChatCommandRegistrant<vetChatCommand>
vetChatCommandReg("!vet",CHATTYYPE_TEAM,0,GAMEMODE_AOW);

```

```

class reb_vet_System : public ScriptImpClass {
void Created(GameObject *obj);
void Custom(GameObject *obj, int message, int param, GameObject *sender);
void Timer_Expired(GameObject *obj, int number);
void Killed(GameObject *obj, GameObject *shooter);
void Destroyed(GameObject *obj);
int Level;
int IconID;
};

```

```

class reb_vet_System_Vehicles : public ScriptImpClass {
void Custom(GameObject *obj, int message, int param, GameObject *sender);
void Timer_Expired(GameObject *obj, int number);
void Killed(GameObject *obj, GameObject *shooter);
void Destroyed(GameObject *obj);
int Level;
int ID;
bool IsPoweredUp;
};

```

```

void MDB_SSGM_Building::Killed(GameObject *obj, GameObject *shooter) {
if (Commands->Is_A_Star(shooter)){
VetUpdate(Get_Player_ID(shooter), 25, Commands->Get_Points(shooter));
}
if (Commands->Get_ID(shooter) != 0) {
if (Settings->LogBuildingKills) {
GameObject *Preset = GetExplosionObj();

```

```

if (!Preset) {
    Preset = shooter;
}
FDSMessage(StrFormat("%s destroyed thanks to %ls
(%s)",Translate_Preset(obj).c_str(),Get_Wide_Player_Name(shooter),Get_Preset_Info(Preset).c_
str()), "_BUILDING");
}
if (Settings->BuildingDeathRewardPage) {
    Console_Input(StrFormat("ppage %d The %s has been destroyed thanks to
you!",Get_Player_ID(shooter),Translate_Preset(obj).c_str()).c_str());
}
if (Settings->GameMode == 1) {
    Commands->Set_Is_Visible(shooter,true);
}
}
}
IsAlive = false;

if (Get_Building_Count_Team(Get_Object_Type(obj)) == 0) {
    BaseControllerClass::Find_Base(Get_Object_Type(obj))->Destroy_Base();
}

}

```

```

void MDB_SSGM_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));
    }
    WasKilled = true;
    if (Find_My_Veh(obj)) {
        Commands->Send_Custom_Event(obj,Find_My_Veh(obj),1111,0,0);
        Console_Input(StrFormat("ppage %d Your vehicle has been unbound because you
died.",Get_Player_ID(obj)).c_str());
    }
    if (Settings->ExtraKillMessages) {
        if (Commands->Get_ID(shooter) == 0) {
            Send_Message_With_Team_Color(Get_Object_Type(obj),StrFormat("%ls was
killed",Get_Wide_Player_Name(obj)).c_str());
        }
        else if (Commands->Get_ID(obj) == Commands->Get_ID(shooter)) {
            Send_Message_With_Team_Color(Get_Object_Type(obj),StrFormat("%ls killed
themselves",Get_Wide_Player_Name(obj)).c_str());
        }
    }
}

```

```

}
else if (!Commands->Is_A_Star(shooter)) {
    Send_Message_With_Team_Color(Get_Object_Type(shooter),StrFormat("The %s killed
%s",Translate_Preset(shooter).c_str(),Get_Wide_Player_Name(obj)).c_str());
}
}
if (Settings->LogPlayerKills) {
    GameObject *ExpObj = GetExplosionObj();
    GameObject *Preset = ExpObj;
    if (!Preset) {
        Preset = shooter;
    }
    if (Commands->Get_ID(shooter) == 0) {
        FDSMessage(StrFormat("%s was killed
(%s)",Get_Wide_Player_Name(obj),Get_Preset_Info(obj).c_str()),"_PLAYERKILL");
    }
    else if (Commands->Get_ID(obj) == Commands->Get_ID(shooter)) {
        FDSMessage(StrFormat("%s killed theirself
(%s)",Get_Wide_Player_Name(obj),Get_Preset_Info(Preset).c_str()),"_PLAYERKILL");
    }
    else if (!Commands->Is_A_Star(shooter)) {
        FDSMessage(StrFormat("The %s killed %s
(%s)",Translate_Preset(shooter).c_str(),Get_Wide_Player_Name(obj),Get_Preset_Info(obj).c_str()
),"_PLAYERKILL");
    }
    else {
        std::string killer = Get_String_Player_Name(shooter),
        killed = Get_String_Player_Name(obj),
        killerpreset = Get_Preset_Info(Preset),
        killedpreset = Get_Preset_Info(obj);
        if (Get_Vehicle(shooter) && !ExpObj) {
            std::string KillMsg =
Settings->VvSKillMsg[Commands->Get_Random_Int(0,Settings->VvSKillMsg.size())];
            StrReplace(KillMsg,"<killer>",killer);
            StrReplace(KillMsg,"<killed>",killed);
            StrReplace(KillMsg,"<killerpreset>",killerpreset);
            StrReplace(KillMsg,"<killedpreset>",killedpreset);
            FDSMessage(KillMsg,"_PLAYERKILL");
        }
        else {
            std::string KillMsg =
Settings->SvSKillMsg[Commands->Get_Random_Int(0,Settings->SvSKillMsg.size())];
            StrReplace(KillMsg,"<killer>",killer);
            StrReplace(KillMsg,"<killed>",killed);
            StrReplace(KillMsg,"<killerpreset>",killerpreset);
            StrReplace(KillMsg,"<killedpreset>",killedpreset);
            FDSMessage(KillMsg,"_PLAYERKILL");
        }
    }
}

```

```
}  
}  
}
```

```
void MDB_SSGM_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));  
    }  
    if (Settings->LogVehicleKills) {  
        GameObject *Preset = GetExplosionObj();  
        if (!Preset) {  
//rest of mdb_ssgm_vehicle::Killed goes here
```

Slipping in the veteran update as you can see is easy, so if you use other versions of ssgm or ssaow then you can easily apply the vetupdate function.

That is partially finished, I wanted to make the veteran points dynamic for buildings, rather than a stock 25 points. I would of done this with getting the player_count, but I have had no time.

The known issue is the SBH and stank getting the keycard attached to them too, which obviously you don't want. You could make a quick solution to this by not attaching the keycards as visual indicators of veterancy at all. But that isn't preferred.

Also, the vehichle veterancy is incomplete, but it is started for you. You just have to add your power-ups to them.

You may also want to add repairing veterancy too.

But you're right Gen, most people that have the ability to share there stuff don't. Or they don't release source code. It is a shame. Are they afraid they will be ridiculed about there code?

I know mine is sloppy as hell. I make errors all the time. There are bound to be people who make fun of it or say it is shit, but fuck em.. We all have to learn.

I've just started a colledge course for my MCSD, so ofcourse I will get better. I won't make so many mistakes in the future and I will look back at code like this and most likely cringe. But we all have to start somewhere.

Start sharing more fuckers.

its not really people getting ridiculed for code more that they don't want other servers to have it.

Its great when people like you like to help other people out and share your work. I hope more people are willing to share there work. isn't that what the mod forums is for ?

The servers that i help mod already has a veteran system because some 1 was nice enough to give it to us, so there still a lot of people who share.

thx reborn