
Subject: Re: Veteran Plugin.

Posted by [reborn](#) on Sat, 24 May 2008 14:44:54 GMT

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

I decided to upgrade the veteran system that I posted here to use in cnc_reborn. I changed it so that it rewards players for healing/repairing and disarming.

I also added some little notices for players too, these notices will only be displayed for people using the upgraded client scripts.dll, but for cnc_reborn that's fine.

Here is the changes I made.

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

std::vector<RepPlayers> RepInfo;

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 RepCheckPoints(int ID) {
```

```

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

```

```

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

```

```

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 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);
}
}
}

```

Then on the ::Damaged events for the vehicles, players and buildings I did the following:

```
void MDB_SSGM_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);
        }
    }
    if (damage > 0.0f) {
        LastDamage = damage;
    }
}
```

```
void MDB_SSGM_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));
        }
    }
}
```

```
if ((IsSecondWind == true) && (Commands->Get_Health(obj) <= 25) &&
(Commands->Get_Health(obj) > 0)) {
    IsSecondWind = false;
    Commands->Set_Health(obj,Commands->Get_Max_Health(obj));
    Commands->Set_Shield_Strength(obj,Commands->Get_Max_Shield_Strength(obj));
}
}
```

```
void MDB_SSGM_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 MDB_SSGM_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: 5");
if (IsDisarmed == false) {
    IsDisarmed = true;
    FDSMessage(StrFormat("%ls disarmed a
%s",Get_Wide_Player_Name(shooter),Translate_Preset(obj).c_str()),"_BEACON");
}
}

```

```

void MDB_SSGM_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: 1");
    if (Settings->LogC4) {
        FDSMessage(StrFormat("%ls %s has been disarmed by %ls (Owner: %ls - Attached to:
%s)",Get_Wide_Team_Name(Get_Object_Type(obj)),Translate_Preset(obj).c_str(),Get_Wide_Pl
ayer_Name(shooter),Get_Wide_Player_Name(Get_C4_Planter(obj)),Translate_Preset(Get_C4_Att
ached(obj)).c_str()),"_C4");
    }
    WasDisarmed = true;
}

```

I added the script to players on the join hook, the same as the veteran thing, here:

```

void Player_Join_Hook(int i,const char *Nick) {
    VetAddPlayer(i);
    RepAddPlayer(i);
}

```

And I clear the info on the game over event here:

```

void GameOver() {
    VetClearPlayers();
    RepClearPlayers();
    char ObjectsType[10],ObjectsType2[10],ObjectsFile[20];
}

```

I also added those little messages on the normal veteran points gained too, here:

```

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));  
Display_Int_Player(shooter,GetPoints(Commands->Get_Preset_Name(obj)),"Veteran points  
gained: %d"),GetPoints(Commands->Get_Preset_Name(obj));
```

```
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));  
    Display_Int_Player(shooter,GetPoints(Commands->Get_Preset_Name(obj)),"Veteran points  
gained: %d"),GetPoints(Commands->Get_Preset_Name(obj));
```

```
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));  
    Display_Int_Player(shooter,25,"Veteran points gained: 25");  
    }  
}
```

Hope someone finds a use for it.. I may even make it a plug-in if I have time in the next few days, maybe...

Also, my maths seems to be fading away from me...

There must be a better way to turn a negative number into a positive number, rather than "(damage - damage) - damage".