
Subject: BUG/TYPO M00_Damage_Modifier_DME
Posted by [Troll King](#) on Mon, 09 Jul 2012 17:37:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

I have found a bug in M00_Damage_Modifier_DME.
A bot cannot kill an object with this script attached. No matter what you fill in for 'Killable_by_NotStar'
(Unless you send a custom of 9037 with param 0 to disable the script)
This is what it says at ScriptRegistrant:
Damage_multiplier:float, Star_Modifier=1:int, NotStar_Modifier=1:int, Killable_By_Star=1:int, Killable_by_NotStar=1:int

But it searches for this:
this->Killable_ByNotStar = ScriptImpClass::Get_Int_Parameter(this, "Killable_ByNotStar");

Can this be fixed or does there need to be made a clone that works?

EDIT: I know that it says Killable_By_Star=1:int. But it wont set it to standard value, because the name is incorrect before anyone says that.

Subject: Re: BUG/TYPO M00_Damage_Modifier_DME
Posted by [Troll King](#) on Mon, 09 Jul 2012 23:27:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Clone of M00_Damage_Modifier_DME but then made multiplayer compatible. Some changes made but effect is the exact same.

Toggle Spoiler

```
class M00_Damage_Modifier_DME : public ScriptImpClass
{
public:
void Register_Auto_Save_Variables();
private:
void Created(GameObject *obj);
void Custom(GameObject *obj, int type, int param, GameObject *sender);
void Damaged(GameObject *obj, GameObject *damager, float amount);
float CurrentHealth;
float Health;
int Killable_By_Star;
int Killable_By_NotStar;
float Star_Modifier;
float NotStar_Modifier;
bool Enabled;
};

void M00_Damage_Modifier_DME::Register_Auto_Save_Variables()
```

```

{
    ScriptImpClass::Auto_Save_Variable(&CurrentHealth, 4, 1);
    ScriptImpClass::Auto_Save_Variable(&Health, 4, 2);
    ScriptImpClass::Auto_Save_Variable(&Killable_By_Star, 4, 3);
    ScriptImpClass::Auto_Save_Variable(&Killable_By_NotStar, 4, 4);
    ScriptImpClass::Auto_Save_Variable(&Star_Modifier, 4, 5);
    ScriptImpClass::Auto_Save_Variable(&NotStar_Modifier, 4, 6);
    ScriptImpClass::Auto_Save_Variable(&Enabled, 1, 7);
}

void M00_Damage_Modifier_DME::Created(GameObject *obj)
{
    Enabled = true;
    Health = Commands->Get_Health(obj);
    Killable_By_Star = Get_Int_Parameter("Killable_By_Star");
    Killable_By_NotStar = Get_Int_Parameter("Killable_by_NotStar");
    Star_Modifier = Get_Float_Parameter("Star_Modifier");
    NotStar_Modifier = Get_Float_Parameter("NotStar_Modifier");
}

void M00_Damage_Modifier_DME::Custom(GameObject *obj, int type, int param, GameObject
*sender)
{
    if(type == 9037) // You can enable/disable it by sending this custom. Works with original script
    aswell.
    {
        if(param == 1)
            Enabled = 1;
        if(!param)
            Enabled = 0;
    }
}

void M00_Damage_Modifier_DME::Damaged(GameObject *obj, GameObject *damager, float
amount)
{
    if(Enabled)
    {
        if(!Star_Modifier && Commands->Is_A_Star(damager) && Killable_By_Star || !NotStar_Modifier
&& !Commands->Is_A_Star(damager) && Killable_By_NotStar)
            Health = Commands->Get_Health(obj); // Nothing going on here just store our health.
        if(Star_Modifier && Commands->Is_A_Star(damager) && Killable_By_Star || NotStar_Modifier
&& !Commands->Is_A_Star(damager) && Killable_By_NotStar)
        {
            CurrentHealth = Commands->Get_Health(obj);
            float NewHealth = Health - CurrentHealth;
            NewHealth = Health - Get_Float_Parameter("Damage_multiplier") * NewHealth;
            if(NewHealth < 1) // When it hit's smaller than 1 health, kill it automaticly.

```

```

    NewHealth = 0;
    Commands->Set_Health(obj, NewHealth);
    Health = NewHealth;
    CurrentHealth = NewHealth;
}
if(Star_Modifier && Commands->Is_A_Star(damager) && !Killable_By_Star || NotStar_Modifier
&& !Commands->Is_A_Star(damager) && !Killable_By_NotStar)
{
    CurrentHealth = Commands->Get_Health(obj);
    float NewHealth = Health - CurrentHealth; // Because of this line it will always keep it's health >
0 (Do NOT replace with amount.)
    NewHealth = Health - Get_Float_Parameter("Damage_multiplier") * NewHealth;
    Commands->Set_Health(obj, NewHealth);
    Health = NewHealth;
    CurrentHealth = NewHealth;
}
}
}
}

```

```

ScriptRegistrant<M00_Damage_Modifier_DME>
M00_Damage_Modifier_DME_Registrant("M00_Damage_Modifier_DME",
"Damage_multiplier:float, Star_Modifier=1:int, NotStar_Modifier=1:int, Killable_By_Star=1:int,
Killable_by_NotStar=1:int");

```

Subject: Re: BUG/TYPO M00_Damage_Modifier_DME
 Posted by [zunnie](#) on Tue, 10 Jul 2012 01:02:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Nice, thx. Good work

Subject: Re: BUG/TYPO M00_Damage_Modifier_DME
 Posted by [iRANian](#) on Tue, 10 Jul 2012 08:21:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

You should give the script your own prefix instead of "M00" and register it as the M00 one so people know you made it.

Subject: Re: BUG/TYPO M00_Damage_Modifier_DME
 Posted by [Troll King](#) on Tue, 10 Jul 2012 10:52:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Meh it's ok like this I didnt really make it I just cloned it. The layout is almost an exact copy of the

original one. besides the if statement at where they are killable is different. JonWil and I couldn't figure out what calculation was going on there. So I had to implent something myself.
