

Forum:[Hack Forums](#)

Thread: [Difference between offsets and netvars](#)

Minisaints - August 19, 2016, 3:15 pm

Just a quick question, what is the difference between offsets and netvars? I'm researching cheating in CS:GO.

Such as like 0x004C is an offset but 0xFC is a netvar?

Thanks for any help.

zNinja - August 19, 2016, 3:29 pm

Offset is probably address of function or global variable that your cheat engine should override.

And netvars, that is something specific only to cs:go, probably in your case "0xFC" is value that you engine should assign to variable at offset 0x004C.

0xDEAD10CC - August 20, 2016, 9:08 am

Offsets are actual offsets, netvars appear to be offsets from an offset to obtain a specific value. I've not much knowledge on CS:GO but from a couple minutes reading that's my best guess. They hardly look like values that you assign however.

sqsgkrldwg - August 20, 2016, 7:25 pm

Hi,

Netvars in csgo are just offsets linked with a network name.

You have 2 types of variables linked with a name in the game :

- Entity properties (Also called network variables, accessed using ServerClass or ClientClass, SendTable, SendProp)
- Type description variables (not networked, accessed using datamap_t and typedescription_t)

For these 2, you have 2 things :

- The basic C class with member variables
- A map that store the relative offsets of member variables of all C classes and their respective network name in C string. (This map support inheritance in classes)

This way, the game is able to network the changes made to one class to all clients, for instance sending a network packet "Network var changed, m_bIsControllingBot of CCSPlayer class in entity index 5 is now a bool value of true".

As the member variables may not have the same offset in client and server, this is how problem is solved.

Edit : Not all offsets are netvars, but netvars are all offsets. If someone in cheating community says "here is the netvar" so this is the offset of the member variable.

Sometimes you have to add the value to the offset of the sendtable, etc ...

Most of time, offsets are absolute values (in relation to the dynamic library). Netvars are relative values.

M3rc - August 21, 2016, 10:45 am

(Aug 20, 2016 07:25 PM)Leffe Wrote: [\[CITE\]](#) Hi,

Netvars in csgo are just offsets linked with a network name.

You have 2 types of variables linked with a name in the game :

- Entity properties (Also called network variables, accessed using ServerClass or ClientClass, SendTable, SendProp)
- Type description variables (not networked, accessed using datamap_t and typedescription_t)

For these 2, you have 2 things :

- The basic C class with member variables
- A map that store the relative offsets of member variables of all C classes and their respective network name in C string. (This map support inheritance in classes)

This way, the game is able to network the changes made to one class to all clients, for instance sending a network packet "Network var changed, m_bIsControllingBot of CCSPlayer class in entity index 5 is now a bool value of true".

As the member variables may not have the same offset in client and server, this is how problem is solved.

Edit : Not all offsets are netvars, but netvars are all offsets. If someone in cheating community says "here is the netvar" so this is the offset of the member variable.

Sometimes you have to add the value to the offset of the sendtable, etc ...

Most of time, offsets are absolute values (in relation to the dynamic library). Netvars are relative values.

...So when you say "C class", I'm assuming you mean struct? Or did you mean to write C++ class?

sqsgkrldwg - August 21, 2016, 12:38 pm

I was talking about how the game implements netvars.

Here it is explained by valve : https://developer.valvesoftware.com/wiki/Networking_Entities

Here is a server-sided example how to read netvars assuming you have a SDK (You can use it to dump netvars but you'll have to modify the code) :

<https://github.com/L-EARN/NoCheatZ-4/blob/master/server-plugin/Code/Misc/EntityProps.h>

If you want to access a netvar given the CBaseEntity pointer, it's typically CBaseEntity + offset ... ecx + offset of member variable.

M3rc - August 21, 2016, 6:26 pm

(Aug 21, 2016 12:38 PM)Leffe Wrote: [\[CITE\]](#) I was talking about how the game implements netvars.

Here it is explained by valve : https://developer.valvesoftware.com/wiki/Networking_Entities

Here is a server-sided example how to read netvars assuming you have a SDK (You can use it to dump netvars but you'll have to modify the code) :

<https://github.com/L-EARN/NoCheatZ-4/blob/master/server-plugin/Code/Misc/EntityProps.h>

If you want to access a netvar given the CBaseEntity pointer, it's typically CBaseEntity + offset ... ecx + offset of member variable.

Ohh, thanks for clearing that up.

I was confused for a second :)

invalid - *August 31, 2016, 4:05 am*

It's like google doesn't exist for this, right?

Netvars are shit anyway since there are better solutions that cover much more in terms of offsets, this was hinted above already.