FilterScripts VCMP 0.4

**Events different from VCMP ones**

onFilterScriptLoad (instead of onScriptLoad)

onFilterScriptUnload( “ onScriptUnload )

**Return Value in FilterScripts**

If you want to pass the call to other filterscripts, return 1. If you do not want to pass the call to other filterscripts return 0.

Except in onPlayerPM, onPickupClaimPicked, onPlayerRequestClass, onPlayerRequestSpawn, onPlayerChat, onLoginAttempt, onPlayerEnteringVehicle, returning null ( this is also the case, when there is no return statement) will pass the call to next filterscript.

In the case of onPlayerCommand, returning 1( or any non-zero integer infact) means the command is found and call will not be passed to next filterscript.

In onScriptLoad, onScriptUnload, onPlayerJoin, onPlayerPart, onPlayerMove, onPlayerHealthChange, onPlayerArmourChange, onPlayerWeaponChange, onServerStart, onServerStop, onVehicleMove, onVehicleHealthChange, onTimeChange, onClientScriptData, the **return value of the function in main.nut also has an effect.** In these cases, the call is passed to the filterscript only if either the return value is null or return value is non-zero. The call will not be passed to the filterscipts if the return value from main.nut is 0. **You should not** create these functions (onScriptLoad,…, onClientScriptData) slot at runtime, through /exec or ‘ConsoleInput’ as these functions might already be created by the plugin and calls to the filterscript events depend on it. ---------(A)

**Order of calling: first in filterscript or main.nut?**

The function onPlayerCommand is called first in filterscript and then in main.nut. The filterscript must return 1 if the function is found/processed by it. On returning 0 from a filterscript, call is passed to next filterscript. If all filterscripts returns 0, then the function is called in main.nut. The return value of this function in main.nut has no effect for onPlayerCommand.

Like onPlayerCommand, all events in (A) are first called in filterscripts and according to the return value of filterscripts it is called or not called in main.nut. Other functions not specified here, it might depend on the order of filterscript plugin or squirrel plugin.

**Server.cfg and location of scripts.**

filterscripts myfilterscript1 myfilterscript2 myfilterscript3

The above line in server.cfg instructs the server to load three filterscripts- myfilterscript1.nut, myfilterscript2.nut and myfilterscript3.nut. These scripts are to be placed in a folder “filterscripts” in server directory. **Do not include ‘.nut’ extension** in server.cfg

**Reserved Plugin Command ID**= **0x2A1A3C4D** (loadfilterscript) and **0x2A1A3C4E** (unloadfilterscript) and **0x2A1A3C4F**(reload filterscript)

The message string of plugin command must be the name of filterscript like in server.cfg (i.e. without nut extension and without ‘filterscripts/’ prefix.

**Features**

**-**Working onClientScriptData in all filterscripts

-Working Timer in all filterscripts

**Restricted Keywords**

The following keywords are not to be used in main.nut, as plugin uses it as slotnames in roottable.

\_FS\_TimerHandle, \_FS\_onFilterScriptLoad, \_FS\_onFilterScriptUnload, \_FS\_onPlayerJoin, \_FS\_onPlayerPart, \_FS\_onPlayerMove, \_FS\_onPlayerHealthChange, \_FS\_onPlayerArmourChange, \_FS\_onPlayerWeaponChange, \_FS\_onServerStart, \_FS\_onServerStop, \_FS\_onVehicleMove, \_FS\_onVehicleHealthChange, \_FS\_onTimeChange, \_FS\_onClientScriptData, \_FS\_onPlayerCommand.

**Delegate of roottable**

The plugin works by creating a delegate of roottable and use two metamethods: \_get and \_newslot. So setting the delegate of roottable to other tables via main.nut will fail the filterscripts.

**Order of loading filterscripts plugin**

Plugin copies all squirrel functions available in the roottable when the squirrel plugin sends the plugin command( this is before loading script). Other plugins sqlite, mysql etc also register their functions when this plugin command is send. So for the filterscripts plugin to copy the functions registered by other plugins, it is required to **load it at the end.** This means in you server.cfg, in the plugins line, ‘filterscripts04rel32’ must be after ‘sqlite04rel32’, ‘mysql04rel32’, etc but not necessarily before ‘squirrel04rel32’. Otherwise, if the filterscript calls functions of other plugins like ‘ConnectSQL’ of sqlite,

The index ‘ConnectSQL’ does not exist

error will be thrown.