EXILE MOD 1.0.3

SERVER OWNER
GUIDE v2.0

infiSTAR Installation Guide on

Created by Super Jerome

TABLE OF CONTENTS

I. Key Information

The basic installation section is this guide is geared for hosted servers (i.e. Host Havoc). If you are trying to do a Windows platform install go HERE. If you are still confused, a hosted server is a server you rent that comes already set up for ARMA 3. Everything other than the installation steps can be applied to any type of ARMA server hosted or not. Plus if you are good enough to make a Windows platform install from scratch you wouldn't be looking at this guide…..would you?

Key Notes:

- The most important thing to remember is backup everything. Have a folder on your PC containing all files (.pbo, etc.) that you upload to your server.

- Monitor your logs (.rpt) on server startup as well as saving them to your PC.

- Backup your database (MYSQL) regularly (at least once a week).

- If you change a script, double and triple check the lines you modified for missing punctuation marks (i.e. commas).

- Never change the scripting language of a mod unless you know what you are doing. For example, if a mod uses "if (isServer)" don't change it to "if (isDedicated)".

Below is a list of programs you will need:

- FileZilla: this program will give you FTP access to your server. With this you will be able to upload and download files from your server directly.

- Pbo Manager: this is the bread and butter program. It will allow you to access all of the pbo files that pertain to ARMA. In layman's terms it is a WinZip for ARMA 3.

- Dash SQF: basic script editor that color coats things to make it easier to understand. It is basically like a advanced notepad. Cool thing is this comes functional out of the box.

- Notepad++ (optional): same as dash, however, requires an add-on in order to view SQF files properly. Has more features than dash at the moment, but harder to setup.

- Eliteness (advanced users): Eliteness is a part of the mikero-tools and is like Dash and Pbo Manager put together. It's usefulness comes into play in that it can check most, not all, scripts for errors before you run them in ARMA.

- WinMerge: this nifty program is helpful when trying to figure out the differences between two different files. It comes in handy when trying to figure out what Battle Eye filters to import from a mod into your own filters (script.txt).

II. Basic Exile Installation

1. Go to the Exile site and download the client and server files.

2. Log into your control panel, **stop** your server and format your database for Exile.
Note: GTX servers come preformatted for Exile.

3. Using your server FTP address log into your server using FileZilla.

4. Drag both @Exile and @ExileServer into your server root directory.

DRAG HERE

4

5. Create a custom command line and place @Exile in mods and @ExileServer in servermods.

| Service Information | Scheduled Tasks | Actions |

Name: |TeamKronik.com| ExileZ Apocalyptic Chernarus|25k|Tow|VG

Connection Info: 199.60.101.18:2312 **Connect**

Query Info: 199.60.101.18:2313

FTP: 199.60.101.18:8821

Started On: 12/20/2016 10:30:34 PM

Players: 0% (0/40)

CPU: 20.26%

Memory: 23.98% (1.44 GB)

ON Auto Refresh

Running | ☉ **Restart** | ■ **Stop** (pid 7860)

| Configuration Files | Commandline Manager | Mod Manager | File Manager | Log Viewer | Steam Update | Current Activity & Stats |

| Re-create MySQL Database | Update to "profiling" Arma 3 build | MySQL Database Info | Revert to Arma 3 v1.62 | Revert to Arma 3 v1.64 | Format Database for Exile Usage [Wipes Database] | Log into PHPMyAdmin |

Commandline Manager

| **Predefined Commandlines** | **Custom Commandlines** |

Don't Restart | ☐ **New** | ⊗ **Cancel**

| Description | Commandline | | |

Description: Exile

Auto Initialize ☑ Starts mission when server starts, instead of when the first player joins. Requires persistent=1 in server config.

disableServerThread ☐ Option to disable the server send messaging thread in case of random crashing

File Patching ☐

Client Mode (Use Server as Headless Client) ☐

Client Connect ☐

Make sure this is enabled

Client Connect IP (Add space at end) [   ]

LoadMissionToMemory ☐ For more advanced users only (as this may cause performance issues on big mission files).

Maximum Memory [6,144.00]

Mods, separate via ; [@Exile]

Network Logging ☐ Enables multiplayer network traffic logging. Not recommended.

No Logs ☐ Disables logging. Required by some mods.

Parameter Filename [   ]

Ranking Filename [   ]

Server Mods [@ExileServer]

Preview | **Save** | **Cancel**

5

6. Create a folder on your PC and name it whichever map your server will run. For this instruction we will use Altis.

7. Inside your Altis folder create two folders and name one backup and other archive.

8. Download Exile map pbo files. DOWNLOAD LINK
   (Chernarus support will added in the future)

9. Place the Exile.Altis pbo in your Altis folder. Create a copy and put it in backup folder.

10. Drag the Exile.Altis pbo into the mpmissions folder.
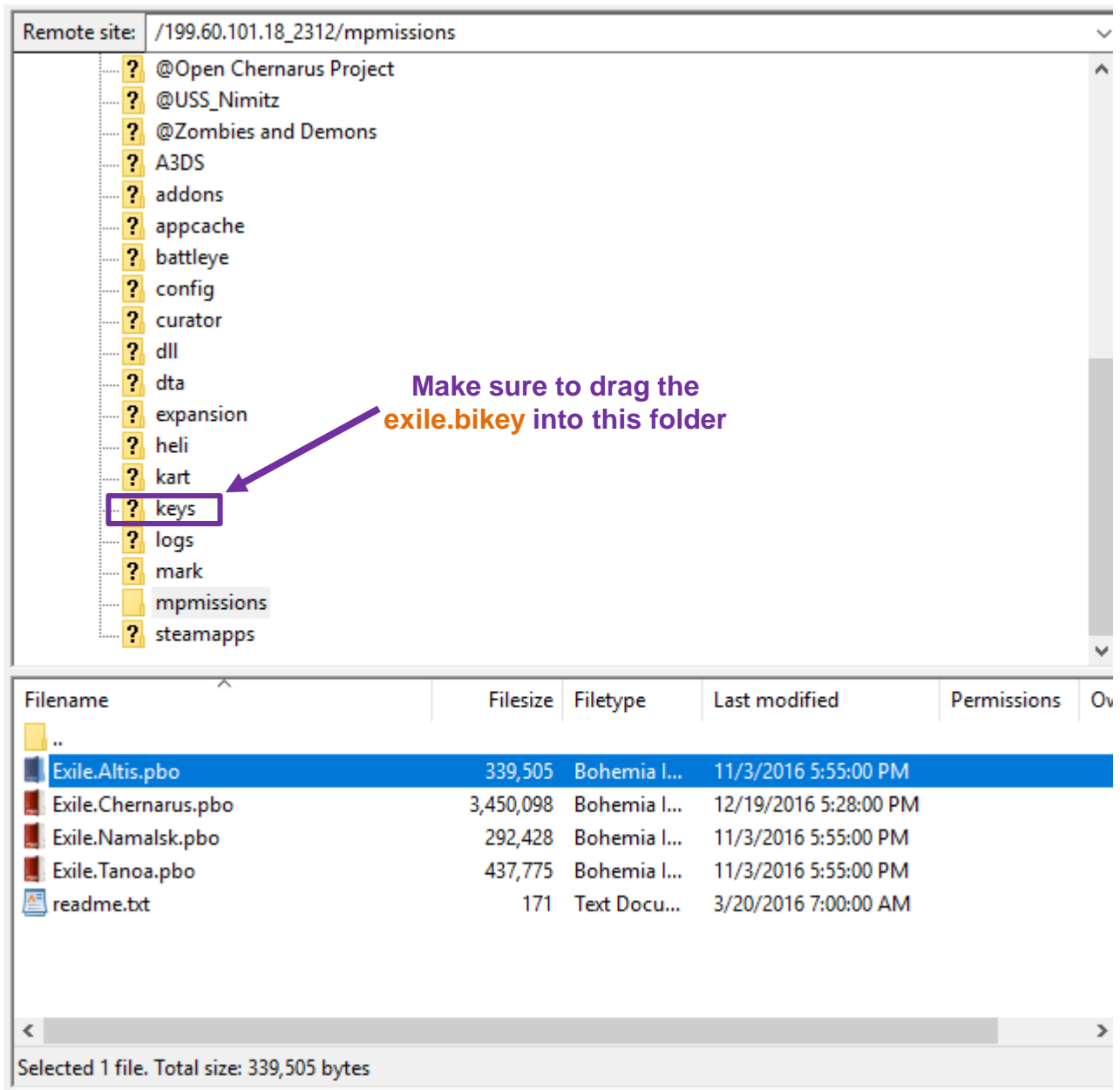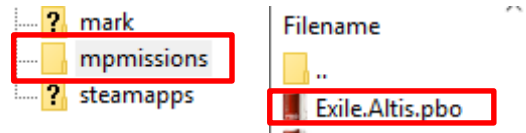
Note: exile.bikey is located inside the key folder on ExileServer download.

11. Prepare to modify your server.cfg file.
Note: This one is a little tricky as the server.cfg file that your server will run depends on your where your command line is pulling it from. To find out where it is being pull from go to your command line. **Sometimes this file is called config.cfg.**

Predefined Commandlines	**Custom Commandlines**

Don't Restart · 🗐 New · ❌ Cancel

Description	Commandline	Edit	Selected
Exile Cherno	-ip=199.60.101.18 -port=2312 -noPause -noSound "-cfg=A3DS\basic.cfg" -ranking="" -par="" "-config=A3DS\server.cfg" "-profiles=A3DS" "-servermod=@m3e_3den;@ExileServer;@marma;@infiSTAR_Exile;@ExileLootDrop;@Advanced_Rappelling;@Advanced_Sling_Loading" "-mod=@Exile;@CBA_A3;@CUP_Terrains_Core;@CUP_Terrains_Maps;@Zombies and Demons;@Extended_Base_Mod;@Arma Enhanced Movement;@Open Chernarus Project;@CUP_Vehicles;@CUP_Weapons;@CUP_Units" -world=empty -autoinit -maxMem=6144	Edit	Selected

|◀ ◀ ▶ ▶| Page size: 10 ▼   1 items in 1 pages

Remote site: /199.60.101.18_2312/A3DS

- ? @CUP_Units
- ? @CUP_Vehicles
- ? @CUP_Weapons
- ? @Exile
- ? @ExileLootDrop
- ? @ExileServer
- ? @Extended_Base_Mod
- ? @infiSTAR_Exile
- ? @m3e_3den
- ? @Open Chernarus Project
- ? @USS_Nimitz
- ? @Zombies and Demons
- ⊞ A3DS
- ? addons
- ? appcache
- ? battleye
- ? config
- ? curator
- ? dll
- ? dta

Filename	Filesize	Filetype	Last modified
arma3server_2016-12-20_19-30-34.rpt	3,012,387	RPT File	12/20/2016 8:30:00 PM
arma3server_2016-12-20_22-30-52.rpt	2,933,975	RPT File	12/20/2016 10:32:00 PM
basic.cfg	1,545	CFG File	11/3/2016 5:55:00 PM
server.cfg	3,074	CFG File	12/19/2016 5:27:00 PM
server_10156.log	14,578	Text Docu...	12/20/2016 11:30:00 A...
server_10412.log	3,960	Text Docu...	12/20/2016 8:29:00 AM
server_10476.log	15,510	Text Docu...	12/19/2016 11:25:00 A...
server_11016.log	21,068	Text Docu...	12/19/2016 2:25:00 PM
server_11080.log	10,877	Text Docu...	12/20/2016 2:30:00 PM

Selected 1 file. Total size: 3,074 bytes

Make sure you are looking at the command line that will run on server start. Different command lines may pull server.cfg from different locations.

12. Open server.cfg in Dash and add Exile.Altis to your mission line.
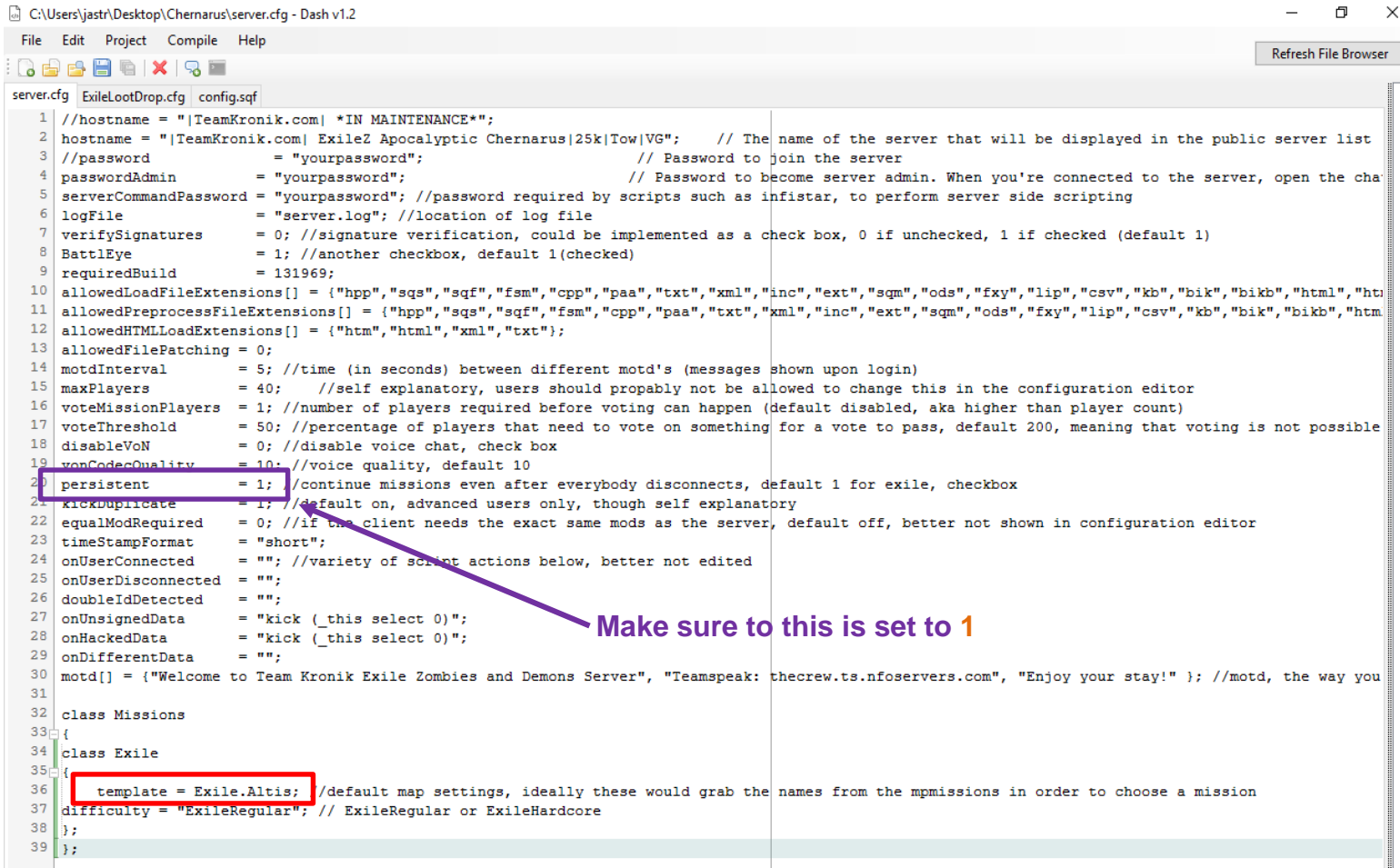
For additional information on server.cfg go HERE.

If you don't have a class Exile in your server.cfg copy and paste this:

```
class Missions
{
class Exile
{
        template = Exile.Altis;  // Never include the ".pbo" in this line
        difficulty = "ExileRegular";  // ExileRegular or ExileHardcore
};
};
```

You are done, now all you have to do is start your server and monitor your logs.

III. Basic Scripting

Before we go any further there has to be an understanding on basic scripting. There will be sometimes where you have to add, remove, or modify a script. At these times it is necessary to understand what you can and cannot do so you don't break your server.

1. **Line Comment** (//): there comes times where you need to omit a line, but don't necessarily want to delete it. Instead of deleting a line you simply put two forward slashes //.

```
//#include "addons_config\A3_vanilla.sqf"
#include "addons_config\CUP.sqf"
#include "addons_config\Exile.sqf"
```

The first line has // at the beginning so when the script is ran ARMA will skip over that line.

- You can also use the // to add some notes at the end of a command. Sometimes mod developers add notes to certain scripts this way.

```
//#include "addons_config\A3_vanilla.sqf" // ARMA standard vehicles
#include "addons_config\CUP.sqf" // Comment this line if you don't use CUP vehicles
#include "addons_config\Exile.sqf"  // Includes the A3_vanilla.sqf as well Exile vehicles
```

The line will be read by ARMA up until the // at that point the game will move on to the next line.

- If you only wanted to run A3_vanilla.sqf without using the Exile.sqf you would simply add // to the beginning of line 3 and remove the // at the beginning of line 1. The mod developer added these notes because running both line 1 and 3 at the same time would cause conflict within the script.

```
#include "addons_config\A3_vanilla.sqf" // ARMA standard vehicles
#include "addons_config\CUP.sqf" // Comment this line if you don't use CUP vehicles
//#include "addons_config\Exile.sqf"  // Includes the A3_vanilla.sqf as well Exile vehicles
```

Now the A3_vanilla.sqf is running and the Exile.sqf will be excluded from the script.

2. **Block Comment** (/* */): much like the line comment the block comment omits lines. The difference is the block comment can be used to omit a series of lines. The start of a block comment is /* and the end of the comment is */. All lines in between the asterisks are excluded from the script.

```
/*
 Exile Occupation by second coming
    http://www.exilemod.com/profile/60-second_coming/
    For support, visit:
    http://www.exilemod.com/topic/12517-release-exile-occupation-roaming-ai-more/
    This script uses the fantastic DMS by Defent and eraser1:
    http://www.exilemod.com/topic/61-dms-defents-mission-system/
*/
```

Some mod developers like to put information or links at the beginning of a script. Block comments allow them the option of doing so without adding a // to every line.

For more information on commenting go HERE

3. **Statements & Blocks**: when it comes to the basics it is only imperative to know what a statement and a block are for punctuation purposes. Forgetting to add a comma or semicolon can make a mod not work or even stop your server from starting. On the other end of the spectrum, we will not be going into incredible detail seeing as this guide is not for actually making a script.

- **Statement**: a statement is a like an instruction. It tells ARMA to execute a command. As a rule there should be only one statement per line (excluding Eden editor). Statements almost always end with a semicolon ; .

- **Block**: a block is a chunk of scripting code grouped together started and ended with curled braces { }. In some cases you may see [ ] used for blocks. All blocks will end with a semicolon ; .

- **Nested Block**: a nested block is set of scripts that lies inside of a block. They usually contain variables or statements. More information on nested blocks further in guide.

```
Statement 1;

Block
{
   Statement 2;

   Nested block
   {
      Statement 3;
      Statement 4;
   };
};
```

In most cases this is how you will see statements and blocks. However, there are some instances where the spaces will be removed. Here is an example:

Nested block { Statement 3; Statement 4; };

The nested block above is the same as the one to the left just with the spaces removed. ARMA will read both the nested blocks the same. Most script writers like spacing it out for ease of reading.

4. **Punctuation**: as stated earlier punctuation can cause your server not to start essentially breaking your server. In fact, 90% of the time your server has a problem starting it is due to some type of punctuation. If you are lucky, your logs (.rpt) will tell you what line you messed up. It is imperative to check over anything you added or changed within a mod. Punctuation is only an issue when you change something as mods typically don't come with incorrect punctuation.

- **Semicolon** (;): a semicolon for ARMA is like ending a sentence. Normally where you would use a period in ARMA you use a semicolon. When you end a statement or a block you put a semicolon. There is one exception however, the only statement that does not get a semicolon is the #include command.

```
#include "config.cpp"          ← #include does not get a semicolon
[] execVM "IgiLoad\IgiLoadInit.sqf";
```

 * *#include* and *[] execVM* are the two most common commands when adding a mod.

#include advanced information HERE.
execVM advanced information HERE.

- **Commas** (**,**): a comma can end your world. Forget to add a comma, server won't start. Add a comma in the wrong place, sever won't start. Put too many commas, server won't start. So how does one determine where to add these things? Let's take a look at some script:

```
SC_staticBandits          = [     // [[pos],ai count,radius,search buildings]
                            [[3768,8928,0],4,200,true],
                            [[7531,5145,0],4,200,true],
                            [[9176,3859,0],4,130,true],
                            [[4985,12516,0],4,130,true]
                            ];
```

Notice the comment at the beginning of block telling us what each variable means for the script

The mod line you see above is for an AI mod called Exile Occupation. The specific line here focuses on the positions of static AI, number of AI, area AI covers, and whether they will search buildings or not (static meaning stationary). Ok lets go a little deeper:

- **Nested block**: here we have a block with several nested blocks on the inside. To be clear this is a nested block: [[3768,8928,0],4,200,true], . As you can see there are four nested blocks within this block and each nested block is separated by a comma. You will notice, however the last nested block does not have a comma. This tells ARMA that there are no more nested blocks left in this block. If you add a comma at the last one you will break the mod as ARMA will be expecting another nested block and there won't be one.

  NOTE: The last nested block NEVER gets a comma.

- **Map Position**: within each nested block there is a map position [3768,8928,0]. A comma separates each position (x, y, z). Sometimes you may see even detailed positions such as [3768.47,8928.89,0.02] the same rules apply don't be fooled by the decimal point.

  More detail on map positions to come in future.

- **Variables**: if you look at the comment within the block it tells us all that we need to know about what each variable is. The map position (pos) for instance is a variable all on it's own. For this script it is contained within two brackets [ ] in most cases this is how you will see map positions. **All variables within a block or nested block are separated by a comma except the last one**. Variables tell the statement what you want to do. For this script, if I wanted to have 10 AI that searched a radius of 300 meters and ignored buildings I would put: [[3768,8928,0],10,300,false]. Let's look at another type of variable:

```
addons[]=
{
   "exile_client",
   "cup_chernarus_config",
   "CUP_Core",
   "Ryanzombies",
   "a3_characters_f"
};
```

No comma on last variable

To ARMA this looks like a single line:
addons[]= {"exile_client", "cup_chernarus_config", "CUP_Core", "Ryanzombies", "a3_characters_f" };

Here we have a block from a mission.sqm, as you can see the variables here look different. Each variable is contained within quotation marks "". Same rules apply.

One thing to remember if you get lost or confused look at the rest of the script you are editing. Follow the punctuation of the rest of the script. In most cases the mod developer will place notes to guide you.

IV. MOD/Script Installation

For this guide we will refer to both mod and script installation as mods since you are actually modifying the normal Exile. This is where most server owners encounter issues (usually ending up with server not starting). To be fair to mod developers this is usually because the server owners don't follow directions. Due to the amount of mods there actually are and the various ways they are installed we are going to go over the basics. We will cover standard clientside mod and serverside mod installation. There are also mods, like Exile, that are both clientside and serverside (combo mods).

1. **Clientside Mods**: a clientside mod is a mod that is installed inside of your map pbo. Earlier the example that was used was the Exile.Altis pbo. We will continue to use this example just keep in mind to apply this to whatever map you are using.

    For this example we will be installing R3F Logistics:

    Step 1: DePbo your Exile.Altis pbo. (This is a multi-step process)

    o Right click Exile.Altis and go to properties
    o Where it says "Opens with" click "Change…"
    o Select PBO Manager from the list of programs.
      ▪ If it is not there hit find more apps. It is located at C:\Program Files\PBO Manager
    o Now you should be able to open Exile.Altis in pbo manager.
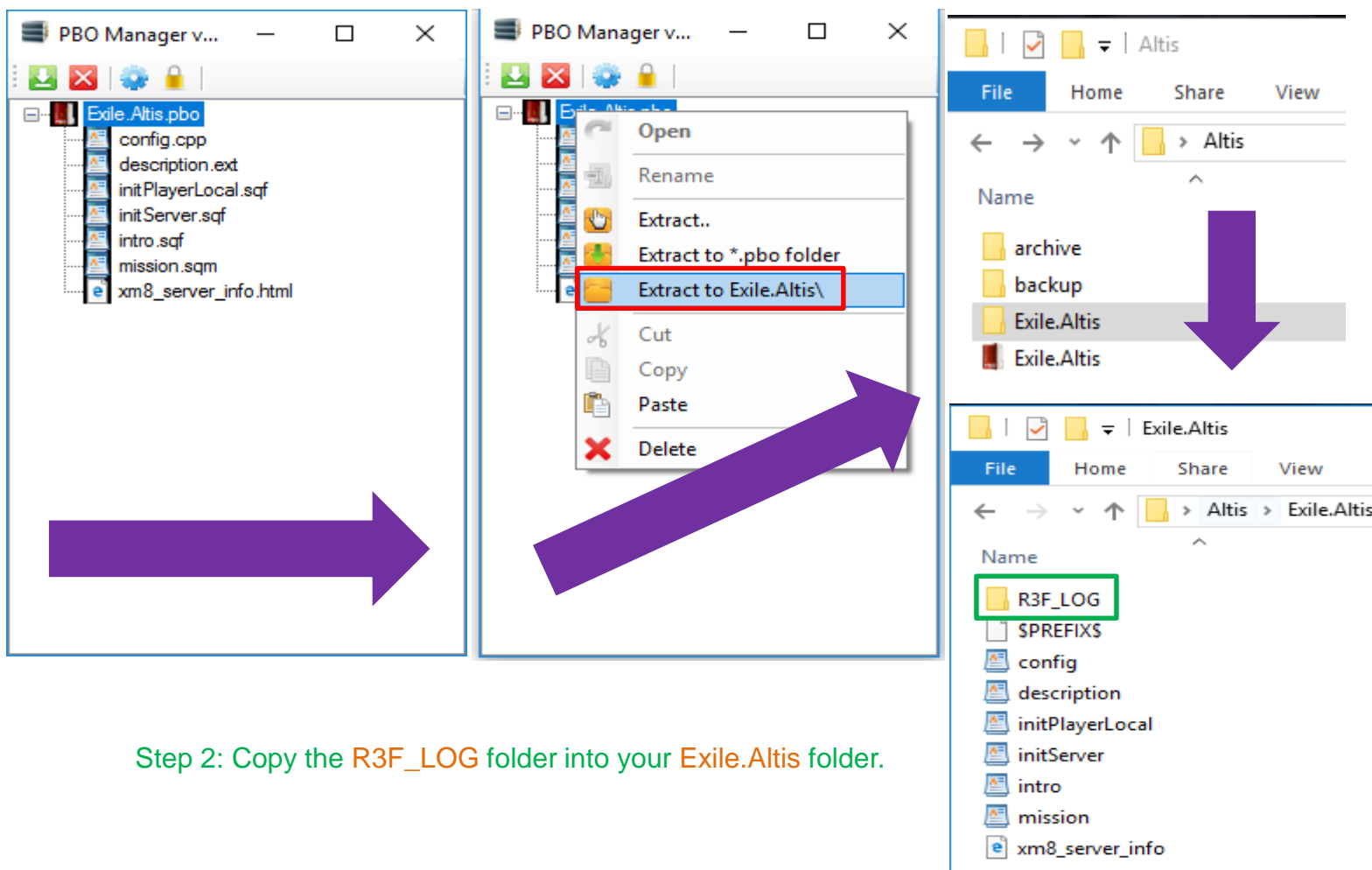
Step 2: Copy the R3F_LOG folder into your Exile.Altis folder.

Just in case you skipped to this section, this is done using Pbo Manager

Step 3: Create an init.sqf and add the execVM statement from R3F Logistics.
- If you don't have an init.sqf you can create one with Dash.
- Create a new file, add the R3F statement, and save it as init.sqf.

Make sure to save in the Exile.Altis folder

Save as an SQF File
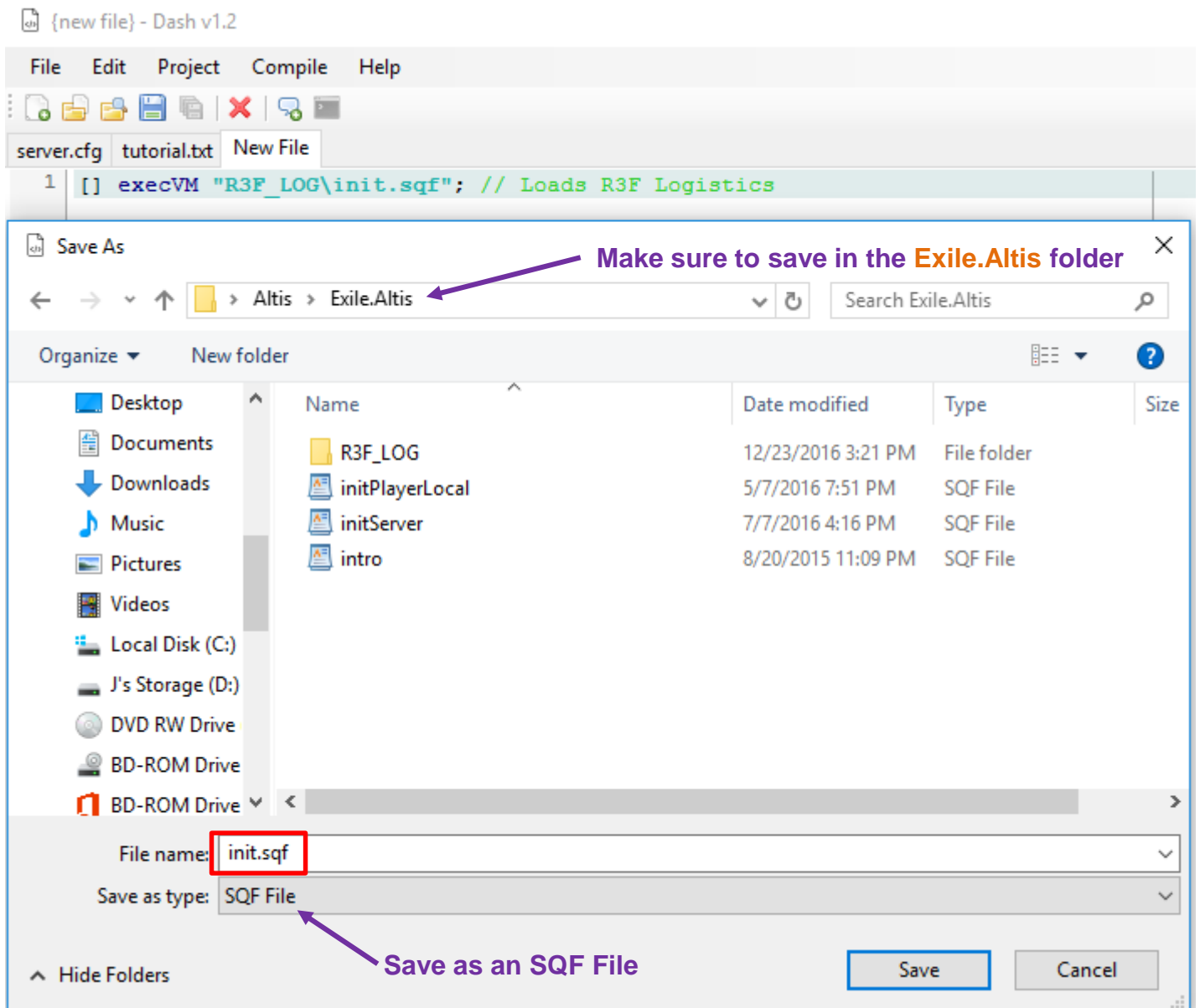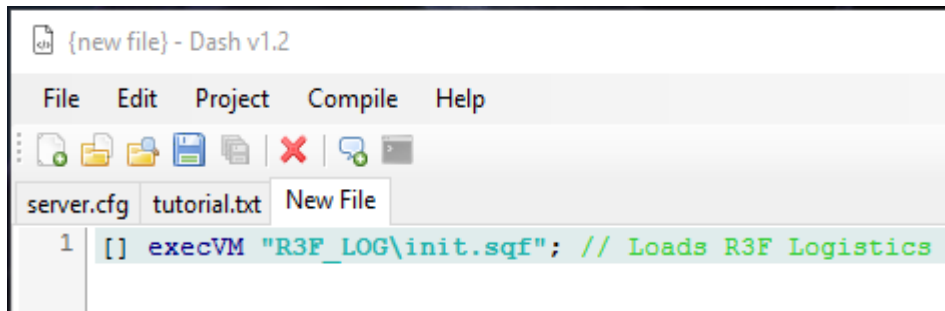
Most clientside mods will require you to add a line to the init.sqf. It helps to leave yourself comments to keep track of the mods you add.

Step 4: Add the R3F statement to your description.ext

- o Make sure the statement is outside of all blocks.

```
54  showHUD[] =
55  {
56      true,    // Scripted HUD (same as showHUD command)
57      true,    // Vehicle + soldier info
58      true,    // Vehicle radar
59      true,    // Vehicle compass
60      true,    // Tank direction indicator
61      false,   // Commanding menu
62      false,   // Group Bar
63      true,    // HUD Weapon Cursors
64      false    // Squad Radar
65  };
66
67  #include "config.cpp"
68  #include "R3F LOG\desc include.h" // Loads R3F Logistics
69
70
71  class CfgRemoteExec
72  {
```

DON'T FORGET TO SAVE

Step 5: RePbo your Exile.Altis and copy it to your mpmissions folder on your server.

**Save the backup created after the RePbo into your backup folder.**
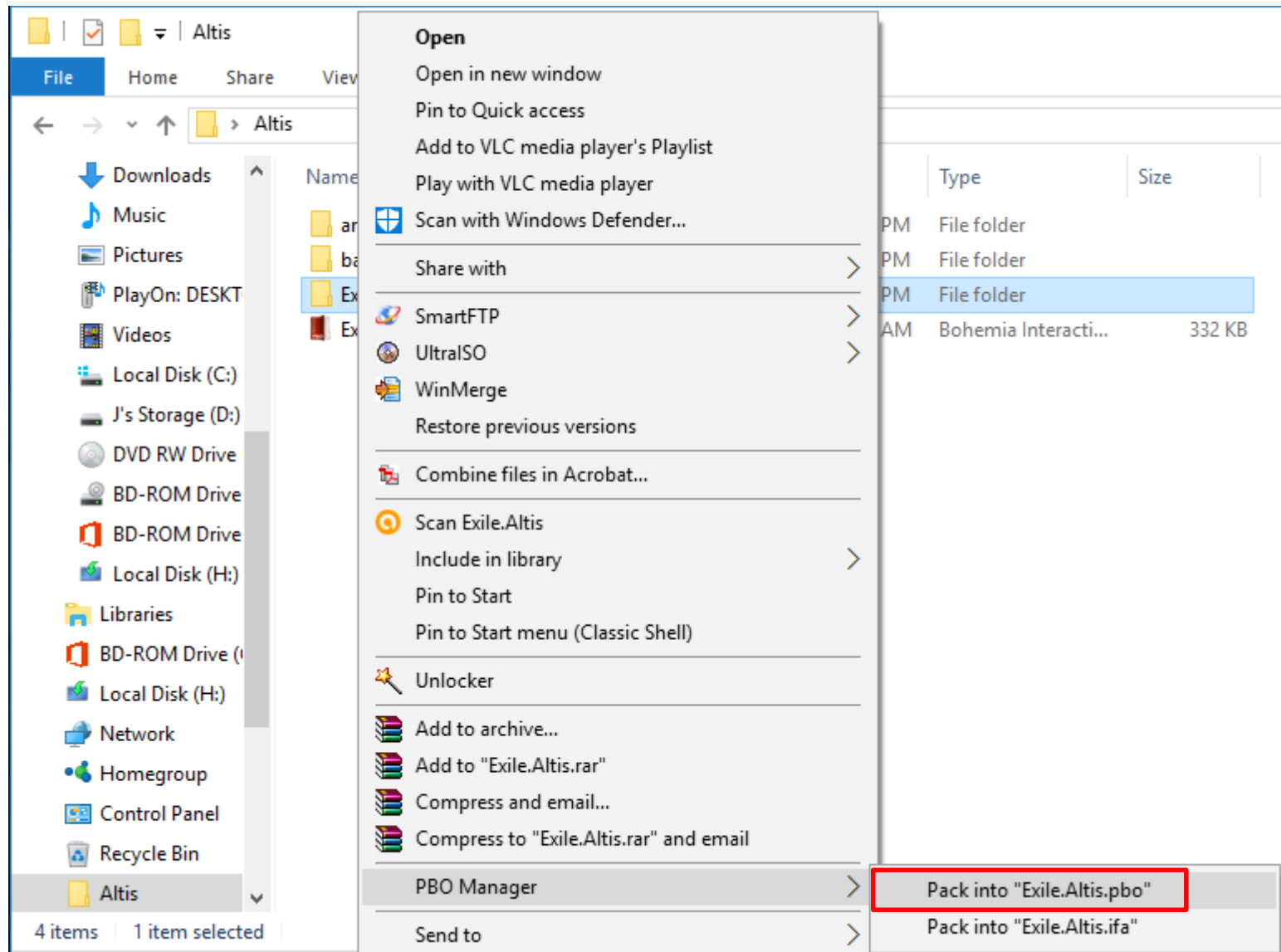
2. **Serverside Mod**: as its name implies these type of mods run serverside. In most cases all this requires is dropping a pbo file into the @ExileServer\addons folder. However sometimes this requires putting the folder in the root of your server and adding a command line. Seeing as the former is simple, here we will be covering the latter. Also, some serverside mods require you to make changes to your MySQL database or your config.cpp inside of your map pbo (i.e. Exile.Altis).
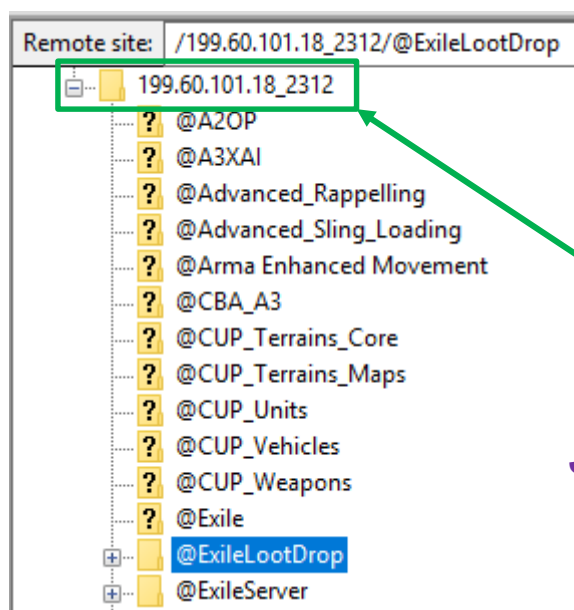
For this example we will be installing Exile Loot Drop:
Note: We will not be modifying the loot tables as they come set up for standard Exile

Step 1: Unzip the download and rename the folder @ExileLootDrop
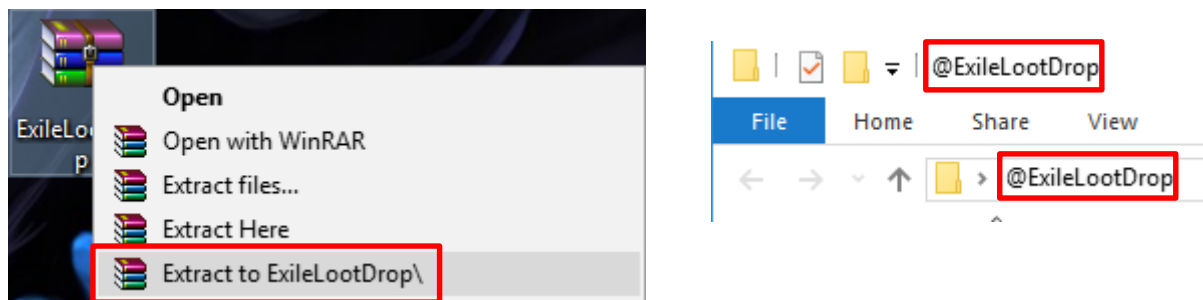
Step 2: Drag the @ExileLootDrop folder into your server root directory.

Just incase you skipped to this section, this is done using FileZilla

Step 3: DePbo your map pbo (i.e. Exile.Altis) located inside your mpmissions folder.

Follow the first step on the clientside mod installation if you need help with this.

Step 4: Go into the Exile.Altis folder open the config.cpp using Dash SQF.

This is an example, use your map pbo

C:\Users\jastr\Desktop\Altis\Exile.Altis\config.cpp - Dash v1.2

File   Edit   Project   Compile   Help

server.cfg   config.cpp

```
2698        class Exile_Car_QilinUnarmed                        { quality = 3; price = 21000; };
2699
2700        ////////////////////////////////////////////////////////////////////
2701        // MB 4WD
2702        ////////////////////////////////////////////////////////////////////
2703        class Exile_Car_MB4WD                               { quality = 2; price = 15000; };
2704        class Exile_Car_MB4WDOpen                           { quality = 2; price = 15000; };
2705
2706        ////////////////////////////////////////////////////////////////////
2707        // Flags
2708        ////////////////////////////////////////////////////////////////////
2709        class Exile_Item_FlagStolen1                        { quality = 1; price = 5000;  sellPrice = 5000;  };
2710        class Exile_Item_FlagStolen2                        { quality = 1; price = 10000; sellPrice = 10000; };
2711        class Exile_Item_FlagStolen3                        { quality = 1; price = 15000; sellPrice = 15000; };
2712        class Exile_Item_FlagStolen4                        { quality = 1; price = 20000; sellPrice = 20000; };
2713        class Exile_Item_FlagStolen5                        { quality = 1; price = 25000; sellPrice = 25000; };
2714        class Exile_Item_FlagStolen6                                             rice = 30000; };
2715        class Exile_Item_FlagStolen7                                             rice = 35000; };
2716        class Exile_Item_FlagStolen8                                             rice = 40000; };
2717        class Exile_Item_FlagStolen9                                             rice = 45000; };
2718        class Exile_Item_FlagStolen10                                            rice = 50000; };
2719  };
2720
2721
2722
2723  class CfgExileCustomCode
2724  {
2725        /*
2726            You can overwrite every single file of our code without touching it.
2727            To do that, add the function name you want to overwrite plus the
2728            path to your custom file here. If you wonder how this works, have a
2729            look at our bootstrap/fn_preInit.sqf function.
2730
2731            Simply add the following scheme here:
2732
2733            <Function Name of Exile> = "<New File Name>";
2734
2735            Example:
2736
2737            ExileClient_util_fusRoDah = "myaddon\myfunction.sqf";
2738        */
2739  };
2740  class CfgExileEnvironment
```

Find

Find: customcode

☐ Match case   ☐ Match whole word   ☐ Regex

Find next   Close

This is a block comment

Create some space in this block after the comment. You will be adding more here most likely.

Step 6: Inside the CfgExileCustomCode block add the two statements from the readme file.

```
2723  class CfgExileCustomCode
2724  {
2725        /*
2726            You can overwrite every single file of our code without touching it.
2727            To do that, add the function name you want to overwrite plus the
2728            path to your custom file here. If you wonder how this works, have a
2729            look at our bootstrap/fn_preInit.sqf function.
2730
2731            Simply add the following scheme here:
2732
2733            <Function Name of Exile> = "<New File Name>";
2734
2735            Example:
2736
2737            ExileClient_util_fusRoDah = "myaddon\myfunction.sqf";
2738        */
2739
2740        //Exile Loot Replacement System
2741        ExileServer_system_lootManager_dropItem = "\ExileLootDrop\ExileServer_system_lootManager_dropItem.sqf";
2742        ExileServer_system_lootManager_spawnLootInBuilding = "\ExileLootDrop\ExileServer_system_lootManager_spawnLootInBuilding.sqf";
2743
2744  };
```

It helps to write yourself comments

Step 7: RePbo your Exile.Altis and upload to mpmissions folder on your server.
Note: Instructions HERE.

Step 8: Add @ExileLootDrop to your command line. Make sure to separate from other mods using a semicolon. Example: @ExileServer;@ExileLootDrop Note: Instructions HERE

3. **Combination Mod**: a combination mod (Yes, I made this name up) is a mod that requires you to install components clientside and serverside. Most mods of this nature are actually serverside, but have menus that players can use which requires a clientside install as well. In some cases these mods also require Battle Eye filters which we will cover later.

For this example we will be installing infiSTAR Exile (AntiHack & Admin Menu):
Notes:
- infiSTAR is a mod that must be purchased first.
- There will not be pictures on things previously covered. Use the Links provided.

Step 1: DePbo your map pbo (i.e. Exile.Altis) and open your description.ext using Dash.
Links: description.ext, DePbo, Dash

Step 2: From the infiSTAR MPMission folder open CfgRemoteExec.hpp using Dash.

C:\Users\jastr\Desktop\infiSTAR.de EXILE_AH v0070 24-Dec-2016 (12-08-19)\MPMission\CfgRemoteExec.hpp - Dash v1.2

File   Edit   Project   Compile   Help

description.ext   CfgRemoteExec.hpp

```
 1  class CfgRemoteExec
 2  {
 3      class Functions
 4      {
 5          mode = 1;
 6          jip = 0;
 7          class fnc_AdminReq { allowedTargets=2; };
 8          class ExileServer_system_network_dispatchIncomingMessage { allowedTargets=2; };
 9      };
10      class Commands
11      {
12          mode=0;
13          jip=0;
14      };
15  };
16
```

Statement/Block Combo

Statements and Blocks explained HERE

It is important to understand the basics of scripting here. You will be merging the class CfgRemoteExec from the description.ext and putting all of the statements/blocks from it into the CfgRemoteExec.hpp. The statements/blocks you will be merging are only the one's that do not exist within the CfgRemoteExec.hpp. Punctuation must be correct as well. Some of the statements you should be looking out for are ones for ExAd XM8 and Advanced Vehicle System (AVS). If you don't use AVS or ExAd then you won't have to worry about this.

Step 3: Merge all statements/blocks from class CfgRemoteExec inside of the description.ext into the CfgRemoteExec.hpp from infiSTAR.
Note: You only need to do this step if you have mods like ExAd and AVS that require statements in CfgRemoteExec. Skip to Step 5 if you have no changes to make.

C:\Users\jastr\Desktop\Chernarus\Exile.Chernarus\description.ext - Dash v1.2

File   Edit   Project   Compile   Help

CfgRemoteExec.hpp   **description.ext**

```
79        #include "ExAdClient\RscTitles.cpp"
80    };
81
82    class CfgHints
83    {
84        #include "ExAdClient\CfgHints.cpp"
85    };
86
87    class CfgNetworkMessages
88    {
89        #include "ExAdClient\CfgNetworkMessages.cpp"
90    };
91
92    class CfgRemoteExec
93    {
94        class Functions
95        {
96            mode = 1;
97            jip = 0;
98
99            class ExileServer_system_network_dispatchIncomingMessage
100           {
101               allowedTargets = 2;
102           };
103           class AVS_fnc_rearmVehicle
104           {
105               allowedTargets = 2;
106           };
107           class AVS_fnc_refuelPayment
108           {
109               allowedTargets = 2;
110           };
111           class AVS_fnc_refuelVehicle
112           {
113               allowedTargets = 2;
114           };
115           class AVS_fnc_fillCanister
116           {
117               allowedTargets = 2;
118           };
119           class ExAdServer_fnc_clientRequest
120           {
121               allowedTargets = 2;
122           };
123       };
124
125       class Commands
126       {
127           mode = 0;
128           jip = 0;
129       };
130   };
```

C:\Users\jastr\Desktop\infiSTAR.de EXILE_AH v0070 24-Dec-2016 (12-08-19)\MPMission\CfgRemoteExec.hpp - Dash v1.2

File   Edit   Project   Compile   Help

CfgRemoteExec.hpp   description.ext

```
1    1class CfgRemoteExec
2    {
3        class Functions
4        {
5            mode = 1;
6            jip = 0;
7            class fnc_AdminReq { allowedTargets=2; };
8            class ExileServer_system_network_dispatchIncomingMessage { allowedTargets=2; };
9
10           class AVS_fnc_rearmVehicle
11           {
12               allowedTargets = 2;
13           };
14           class AVS_fnc_refuelPayment
15           {
16               allowedTargets = 2;
17           };
18           class AVS_fnc_refuelVehicle
19           {
20               allowedTargets = 2;
21           };
22           class AVS_fnc_fillCanister
23           {
24               allowedTargets = 2;
25           };
26           class ExAdServer_fnc_clientRequest
27           {
28               allowedTargets = 2;
29           };
30       };
31       class Commands
32       {
33           mode=0;
34           jip=0;
35       };
36   };
37
```
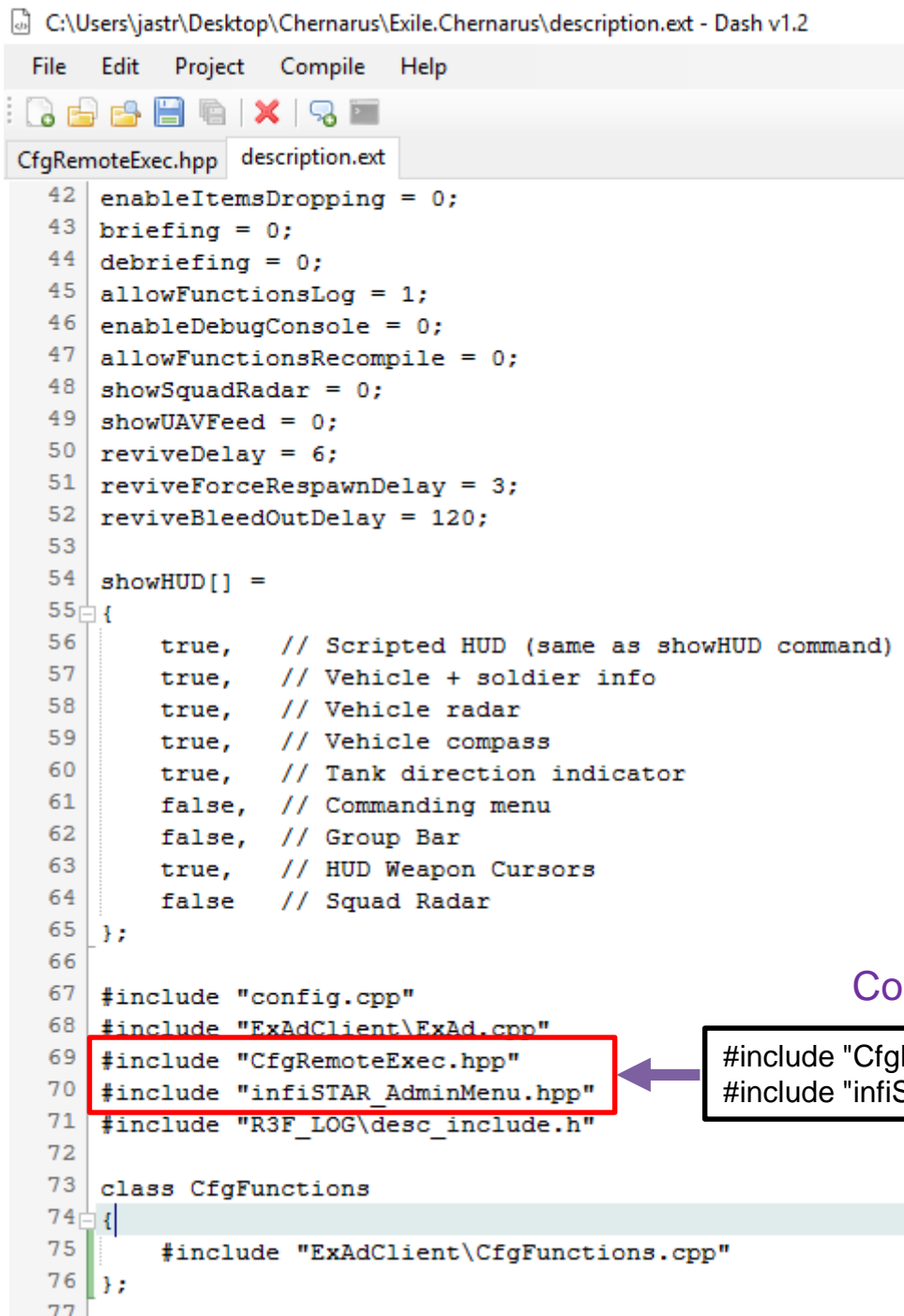
These are the same statement

All these statements go inside of the nested block class Functions. Make sure you check your semicolons

Step 5: Delete the entire class CfgRemoteExec from description.ext

EXILE

Step 6: Inside of description.ext outside of all blocks, put #include "CfgRemoteExec.hpp" underneath this you also put #include "infiSTAR_AdminMenu.hpp".

C:\Users\jastr\Desktop\Chernarus\Exile.Chernarus\description.ext - Dash v1.2

File   Edit   Project   Compile   Help

CfgRemoteExec.hpp   description.ext

```
42  enableItemsDropping = 0;
43  briefing = 0;
44  debriefing = 0;
45  allowFunctionsLog = 1;
46  enableDebugConsole = 0;
47  allowFunctionsRecompile = 0;
48  showSquadRadar = 0;
49  showUAVFeed = 0;
50  reviveDelay = 6;
51  reviveForceRespawnDelay = 3;
52  reviveBleedOutDelay = 120;
53
54  showHUD[] =
55  {
56      true,    // Scripted HUD (same as showHUD command)
57      true,    // Vehicle + soldier info
58      true,    // Vehicle radar
59      true,    // Vehicle compass
60      true,    // Tank direction indicator
61      false,   // Commanding menu
62      false,   // Group Bar
63      true,    // HUD Weapon Cursors
64      false    // Squad Radar
65  };
66
67  #include "config.cpp"
68  #include "ExAdClient\ExAd.cpp"
69  #include "CfgRemoteExec.hpp"
70  #include "infiSTAR_AdminMenu.hpp"
71  #include "R3F_LOG\desc_include.h"
72
73  class CfgFunctions
74  {
75      #include "ExAdClient\CfgFunctions.cpp"
76  };
77
```

Copy and Paste this:

```
#include "CfgRemoteExec.hpp"
#include "infiSTAR_AdminMenu.hpp"
```

Step 7: Copy all files from the MPMission folder inside of the infiSTAR download into your Exile.Altis (or whatever your map is). RePbo your Exile.Altis and upload to your server.

You are officially done with the clientside portion of infiSTAR

Step 8: Configure your EXILE_AHAT_CONFIG to your desires using Dash. This file is located inside of the SERVER_ARMA3_FOLDER\@infiSTAR_Exile\addons\a3_infiSTAR_Exile folder
Note: Your Steam ID should already be in here after purchase. Look at ALL the settings here before moving on especially the serverCommandPassword (has to be same as server.cfg).

Step 9: Once you are satisfied with your settings put the folder a3_infiSTAR_Exile into a pbo.

Right click the folder and you should see this option if you have Pbo Manager.
Once you have it into a pbo you can delete the folder

Step 10: Drag the folder @infiSTAR_Exile to your server root directory.
Note: This is done using FileZilla. Picture instructions HERE.

Step 11: Go into your database (MYSQL) and execute the following command:
Note: This is a multi-part step for owners that have control panel access to their server.

```
/* FOR "USE_DATABASE_WHITELIST" YOU NEED TO RUN THIS QUERY: */
ALTER TABLE `account` ADD `whitelisted` INT(1) UNSIGNED NOT NULL DEFAULT '0';



/* FOR "USE_LOG_TO_DATABASE" YOU NEED TO RUN THIS QUERY: */
CREATE TABLE `infistar_logs` (
          `id` INT(11) NOT NULL AUTO_INCREMENT,
          `logname` CHAR(50) NOT NULL,
          `logentry` TEXT NOT NULL,
          `time` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
          PRIMARY KEY (`id`)
)
COLLATE='latin1_swedish_ci'
ENGINE=InnoDB
AUTO_INCREMENT=1
;
```

Step 12: Using FileZilla go into your @ExileServer\extDB\sql_custom_v2 folder and copy your severs exile.ini to your PC.

Step 13: Open your exile.ini using Dash and add the following lines to the bottom.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;          Author: Chris(tian) "infiSTAR" Lorenzen
;;          Contact: infiSTAR23@gmail.com // www.infiSTAR.de
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;          please add everything below this comment to the bottom of your exile.ini file
;;          to allow your exileserver/infiSTAR to communicate with your database!
;;
;;          path where exile.ini should be:
;;          Arma 3\@ExileServer\extDB\sql_custom_v2
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

[getAccountWhitelisted]
SQL1_1 = SELECT whitelisted FROM account WHERE uid = ?
Number Of Inputs = 1
SQL1_INPUTS = 1
OUTPUT = 1

[insert_infiSTARLog]
SQL1_1 = INSERT INTO infistar_logs SET logname = ?, logentry = ?
Number Of Inputs = 2
SQL1_INPUTS = 1,2
```

Once you **SAVE** this file in Dash you can upload it back to your server.

Step 14: Drag the ARMA_LOAD.dll and ARMA_LOG.dll from the SERVER_ARMA3_FOLDER to you server root directory.
Note: Go HERE if you need a picture of your root directory.

Step 15: Using WinMerge open and compare your own script.txt with the script.txt from the infiSTAR folder. Make any necessary changes and upload back to your BattleEye folder.
Note: You more than likely won't need to make any changes here, but check anyhow.

Final Step: Add @infiSTAR_Exile to your servermod command line and start your server.
Note: Make sure to separate from any other mods using a semicolon:
@ExileServer;@infiSTAR_Exile

V. COMMAND LINE & CLIENT/SERVER MODS EXPLAINED

We have covered basic mod installation already. This section is geared for explaining what makes a clientside mod for clients and serverside mods for servers. Also in this section, we will be going over what determines where you put the @Mod (i.e. @CBA_A3) command at in your command line. For instance Exile has two commands @Exile and @ExileServer.

1. **Clientside Mods**: not all clientside mods actually go inside of your map pbo (i.e. Exile.Altis). Some actually are on the server. What makes a mod clientside is one or more of the following:

   - Has menus that allow the player to do things not standard to ARMA (i.e. R3F Logistics)
   - Contains weapons, vehicles, or other items not standard to ARMA (i.e. CUP_Vehicles)
   - Adds effects like fire, dust, smoke or sounds not standard to ARMA (i.e. RyanZombies)
   - Adds buildings or other terrain features that are not standard to ARMA (i.e. CUP_Terrains)

Simply put clientside mods effect players directly. Giving them features not standard to Exile. There are some rare instances like with Advanced Sling Load where you can run them serverside, but for the most part all mods in this category must be clientside.
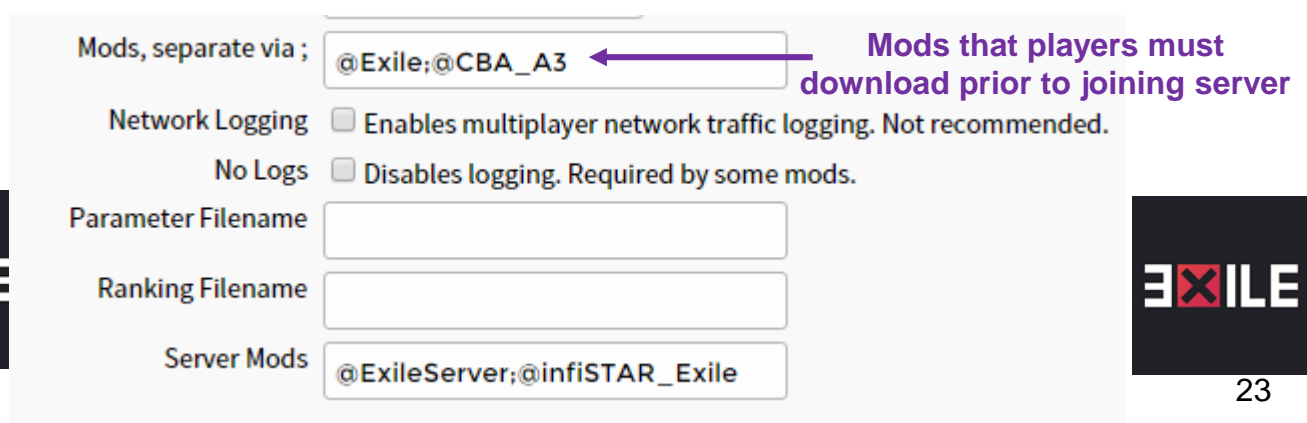
   A. **Map Pbo**: the clientside mods that go inside of your map pbo (i.e. Exile.Altis) are downloaded to players PC when they connect to the server. Having this said. **Be careful not to put any passwords or other confidential information into your map pbo.**

   B. **Download / Subscribe**: These mods require players to download or subscribe to the mods prior to joining the server. Mods of this nature are usually signed by the developer requiring server owners to also install the key that comes with them. You will also need to add a command to your command line instructing your server to run the mod.
   Note: Keys and signatures will be explained in detail later in guide.

2. **Serverside Mods**: mods in this category directly effect the server itself or use features that are contained within standard ARMA or Exile. For instance Bigfoots Shipwrecks uses a crate that is within Exile. Even though this crate is not standard to ARMA, Exile is already running clientside so we are able to place this mod on the serverside and pull from Exile features. Most serverside mods can simply be placed in the @ExileServer\addons folder and will be run automatically. In some cases, however, you must load these mods in your server root directory and add a command to your command line. The mod developer will instruct you on how to install these type of mods.

3. **Command Line**: once you have determined whether the mod you wish to install is clientside or serverside you can determine where to add it at in your command line. Keep in mind not all mods go into your command line. In most cases it will be clear whether to add a mod to your command line and where it goes at in the command line. The most common cases where this won't be explained is usually when you subscribe to a mod on the Steam Workshop. When determining how to install a mod off of Steam use the instructions above. Remember, when adding a mod to your command line to separate each mod using a semicolon.

Mods, separate via ;	@Exile;@CBA_A3 ← **Mods that players must download prior to joining server**
Network Logging	☐ Enables multiplayer network traffic logging. Not recommended.
No Logs	☐ Disables logging. Required by some mods.
Parameter Filename	
Ranking Filename	
Server Mods	@ExileServer;@infiSTAR_Exile

VI.   SERVER SECURITY: SERVER.CFG AND KEYS

1. **Server.cfg**: your server.cfg file is like the brain of your server. It is also the central nervous system of your server's security parameters. An incorrectly setup server.cfg file can allow hackers free reign on your server or, on the other spectrum, kick/ban players that are not hackers. It is necessary to find the balance within your server.cfg that way you can make hackers lives as most difficult as possible while still allowing normal players to play kick free.

   A. **BattlEye**: BattlEye is a global anti-cheat system that is for all ARMA servers that have it enabled. If a player gets banned from one server they are banned from them all. In your server.cfg this is enabled by the command Battleye = 1.

   B. **Extensions**: there are 3 commands that fall in this category allowedLoadFileExtentions, allowedPreprocessExtensions, and allowedHTMLLoadExtensions. For your server, extensions can potentially give a player full access to server or allow other exploits. Not listing any extension means everything is allowed. If too strict, however, then the server's log file will contain warning entries about unable to read. I will give you the best loadout possible (according to Bohemia) that will give you good security but not be too strict. Use the following loadout for your extensions:

```
allowedLoadFileExtensions[] =
{"hpp","sqs","sqf","fsm","cpp","paa","txt","xml","inc","ext","sqm","ods","fxy","lip","csv","kb","bik","bikb","html","htm","biedi"};

allowedPreprocessFileExtensions[] =
{"hpp","sqs","sqf","fsm","cpp","paa","txt","xml","inc","ext","sqm","ods","fxy","lip","csv","kb","bik","bikb","html","htm","biedi"};

allowedHTMLLoadExtensions[] = {"htm","html","xml","txt"};
```

   C. **Signatures**: when it comes to security signatures plays the most important role. There are things neither BattlEye or extensions will catch and that is pbo modification. A player could modify a normal Exile pbo and enable it so that they could give themselves poptabs at will. Signatures would detect a modified pbo and kick/ban that player from the server. However, this is where the balance need to be. A too strict signatures setup will kick normal players from your server. Signatures is effected by the following settings in your server.cfg:

      1) **verifySignatures**: this command is the heart of the signatures security settings. With this disabled none of the other settings will have an effect on the server. There are two versions of this v1 (= 1) and v2 (= 2). As of ARMA patch 1.59, v1 is obsolete so if you want this enabled you put verifySignatures = 2; (disabled is = 0).

      2) **onHackedData**: what this and the following commands have in common is the actions that the server will take if it finds a pbo that has been altered. This particular command actually detects modifications of a signed pbo. You can either ban or kick players with this command. Bohemia and infiSTAR recommends the kick due to the command's sporadic nature. The command for kick is (to ban replace kick with ban): onHackedData = "kick (_this select 0)";
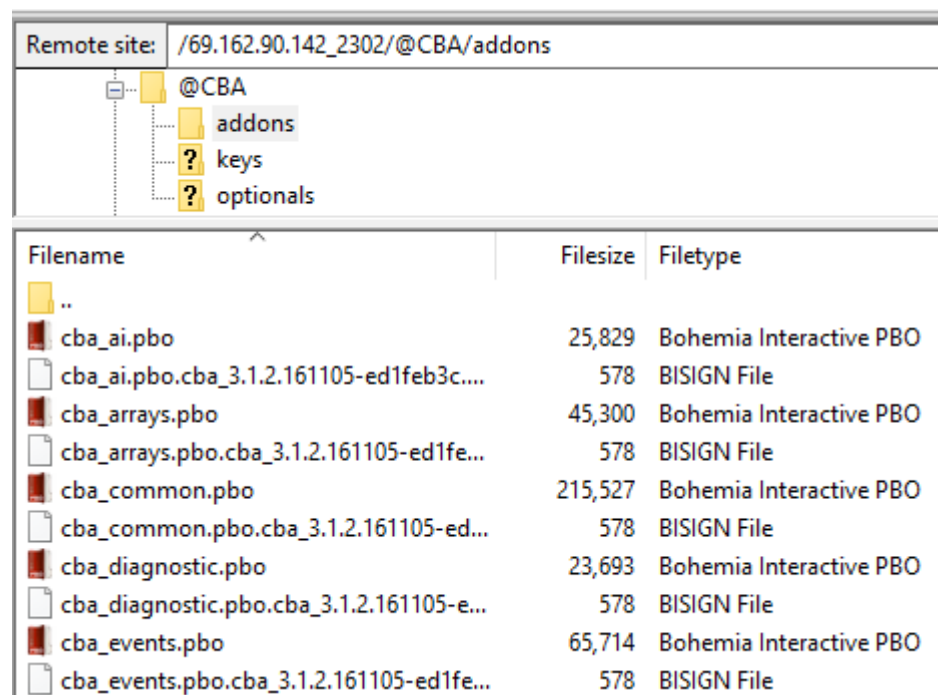
3) **onUnsignedData**: basically what this command does is if a player joins your server with a pbo that is supposed to be signed it will kick/ban them. This is good to have because someone could change a pbo then delete the BISIGN file associated with it and join your server. The command for this should look like this:
onUnsignedData = "kick (_select 0)";
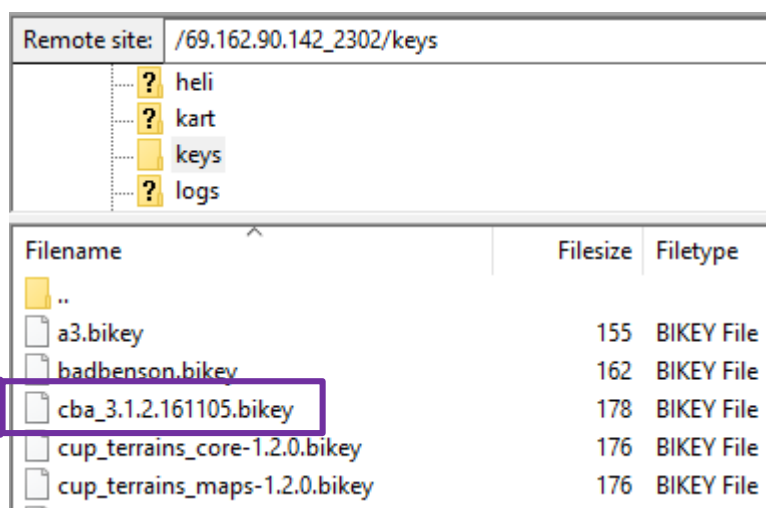Note: BISIGN files will be explained later in guide

4) **onDifferentData**: this command is the devil incarnate. If you have ever been randomly kicked off of a server this is the reason why. It is supposed detect an outdated pbo. What it actually does is kick a player with only a different, but valid pbo. In Bohemia's own words "very strict test, use sparingly, no longer supported in ARMA 2 OA 95232". If you like having players on your server you will set this command to: onDifferentData = "";
Note: Bohemia's information link HERE

2. **Keys**: when a Bohemia or a server developer makes a pbo they use a key to sign it. These signatures are called BISIGN files and are what verifySignatures actually check when it verifies a players files. Our jobs as server owners is to ensure that we have the most up to date keys and BISIGN files on our server. Seeing as signatures and keys are to verify clientside files, you will not see keys on a serverside mod. Below are some pictures of keys and BISIGN files.

This one CBA key is linked to all of these BISIGN files. The BISIGN files are linked to the pbo's they are named after.

SAMPLE SERVER.CFG
Don't forget to set ALL passwords

```
hostname = "My Server Name"; // put your server name here
//password = "yourpassword"; // If public server leave this alone
passwordAdmin        = "yourpassword"; // when in game, open the chat and type: #login password
serverCommandPassword = "yourpassword"; // has to match infiSTAR pbo EXILE _AHAT_CONFIG
logFile            = "server.log"; // location of log file
verifySignatures      = 2; // either 2 for enabled or 0 for disabled
BattlEye            = 1; // BattlEye on is 1 and off is 0
requiredBuild        = 139586; // current build as of ARMA update 1.66
allowedLoadFileExtensions[] =
{"hpp","sqs","sqf","fsm","cpp","paa","txt","xml","inc","ext","sqm","ods","fxy","lip","csv","kb","bik","bikb","ht
ml","htm","biedi"};
allowedPreprocessFileExtensions[] =
{"hpp","sqs","sqf","fsm","cpp","paa","txt","xml","inc","ext","sqm","ods","fxy","lip","csv","kb","bik","bikb","ht
ml","htm","biedi"};
allowedHTMLLoadExtensions[] = {"htm","html","xml","txt"};
allowedFilePatching = 0; // not needed for Exile
motdInterval = 5; // doesn't effect anything in Exile
maxPlayers = 40; // whatever your server's max players is Exile max is 70
voteMissionPlayers = 200; // leave this alone or players can vote change your map
voteThreshold = 50; // percentage of players that need to vote on something for a vote to pass
disableVoN = 0; // disable voice chat, check box
vonCodecQuality = 20; // voice quality, default values are 1-30, recommend 20
persistent = 1; // this must be 1 for Exile (DON'T CHANGE)
kickDuplicate = 1; // default on, advanced users only, though self explanatory
equalModRequired = 0; // NEVER TURN THIS ON IT IS OUTDATED
timeStampFormat = "short";
onUserConnected = ""; // don't touch this unless you know what you are doing
onUserDisconnected = ""; // don't touch this unless you know what you are doing
doubleIdDetected = ""; // don't touch this unless you know what you are doing
onUnsignedData = "kick (_this select 0)";
onHackedData = "kick (_this select 0)"; // recommend kick over ban
onDifferentData = ""; // to strict will kick normal players, like having players leave alone
motd[] = {"Your motd here." }; // motd, the way you usually see this done is a text box with an add line
button to add a second line (the comma separated strings in this config file)
class Missions
{
class Exile
{
template = Exile.Altis; // put your map here, do not include the .pbo
difficulty = "ExileRegular"; // ExileRegular or ExileHardcore
};
};
```

Note: The extensions lines are only 1 line. DO NOT MAKE TWO LINES FOR THEM

Information on finding location of server.cfg HERE

VII. BATTLEYE FILTERS

Note: Before you read any further, this section has nothing to do with creating your own filter. It is about applying a mod's filter into your own scripts.txt file. I will list some links to guides for this at the end of this section.

BattlEye can be the most aggravating thing about making a server. You install some awesome mod, join your server and then boom kick/ban. The key to making sure this does not happen is when you are installing a mod to install the BattleEye filter that comes with it. In most cases when you are installing a mod, the mod developer actually gives you the filter for it. In other cases all you get is a script.txt and then you are stuck trying to figure out what to add. We will cover both instances so that you can keep those random kicks to a minimum.

1. **Adding a filter to your script.txt**: after you get your mod installed the server developer gives you a filter that looks like this:

> 7 deleteVehicle !="Exile_Unit_Player" !"NVG_Target" !"Preview"

But when you get to your script.txt there is no line for 7 deleteVehicle. You would simply add a the line and then right after it you put your filter simple as that. Now if you already have the line you would just put the filter at the beginning of line 7 deleteVehicle. Here are some quick steps.

Step 1: Obtain your script.txt using FileZilla from your BattlEye folder.
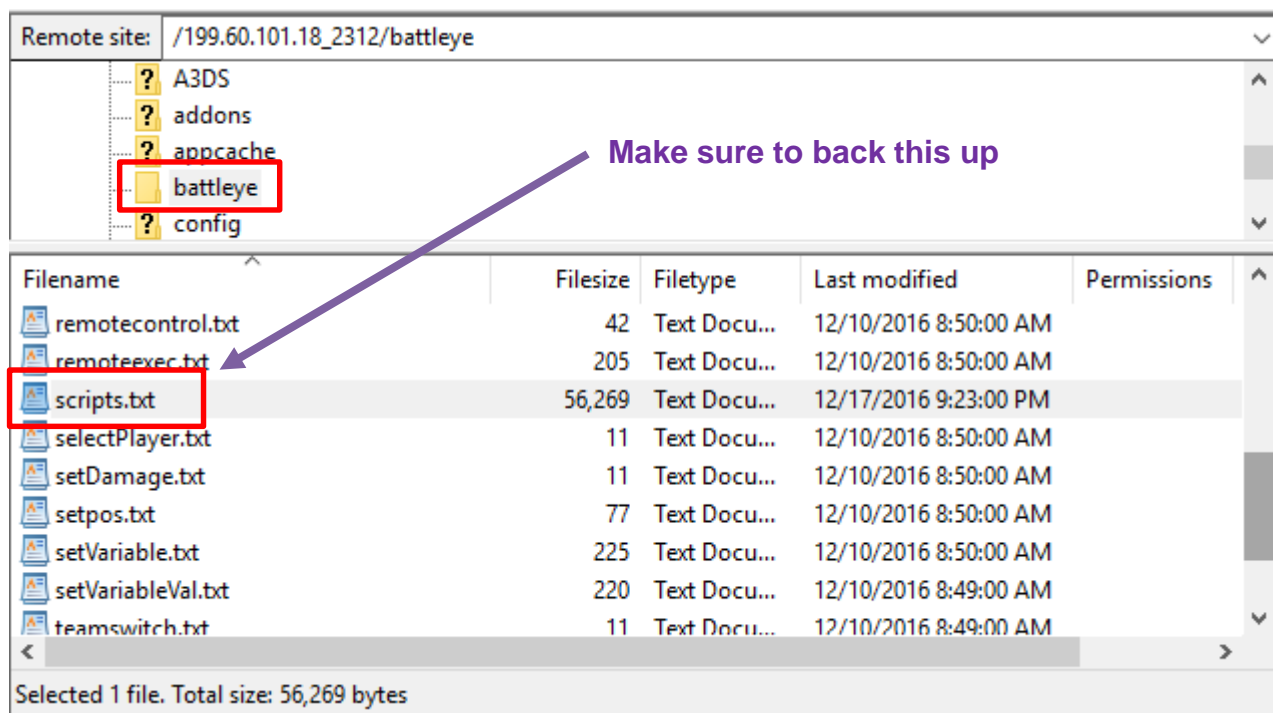
Step 2: Open your script.txt using Dash
Step 3: Add a the line 7 deleteVehicle (if it is not there) and put the filter at the beginning.
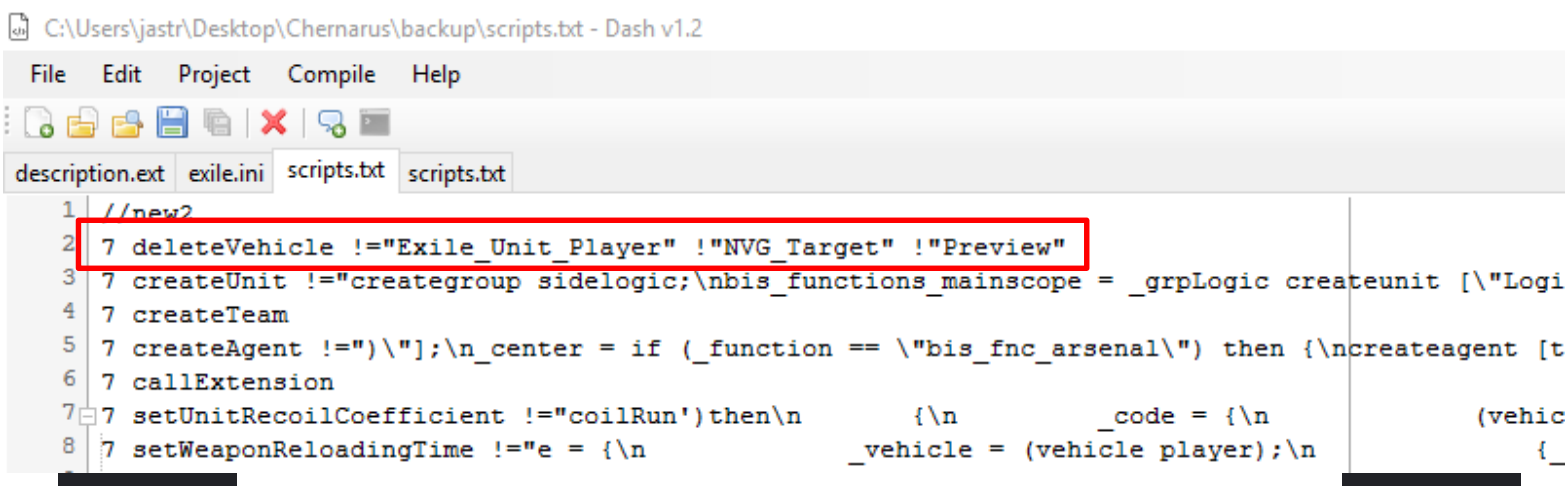
**Upload this back to your BattlEye folder and you are done.**

2. **Comparing your script.txt to another**: if you happen to get a script.txt from a mod like ExAd or just another person then things get a little bit more complicated. You then have to determine what filters are in the script.txt from the mod that are not in yours. Originally I have been using WinMerge, but noticed some flaws in it. So instead you will use Notepad++ and install some plugins to make it work properly with BattlEye filters. You will need to install a plugin first in order to be able to compare files within Notepad++. Follow the below steps:

    Note: Optional addon for SQF language HERE

    Step 1: With Notepad++ open click on Plugins > Plugins Manager > Show Plugins Manager
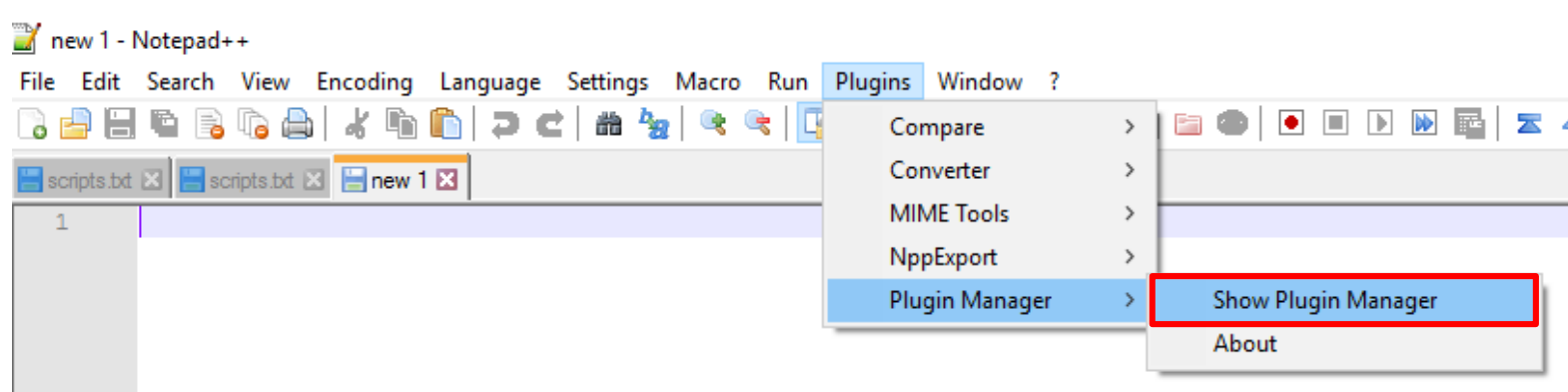
    Step 2: Once inside the Plugins Manager find the plugin Compare check and install it.
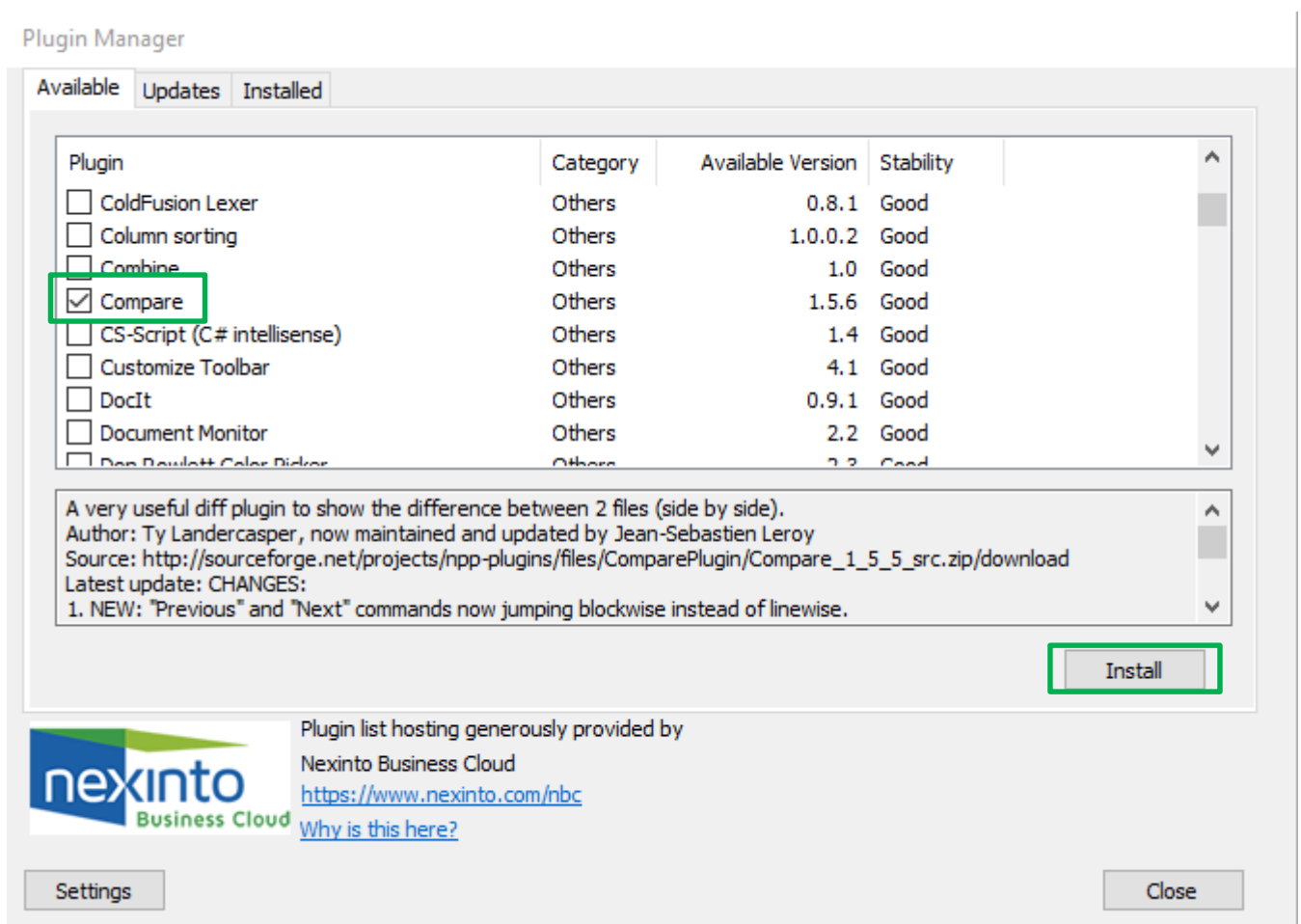
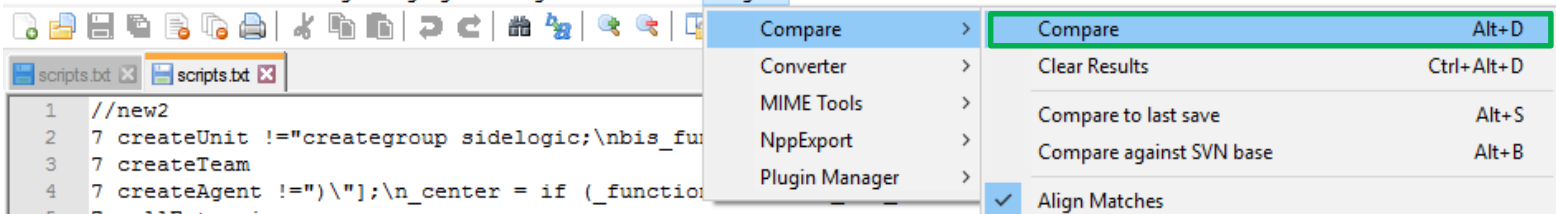

Note: You will get two message boxes just hit Yes on both.

Step 3: Open the two script.txt files that you wish to compare.
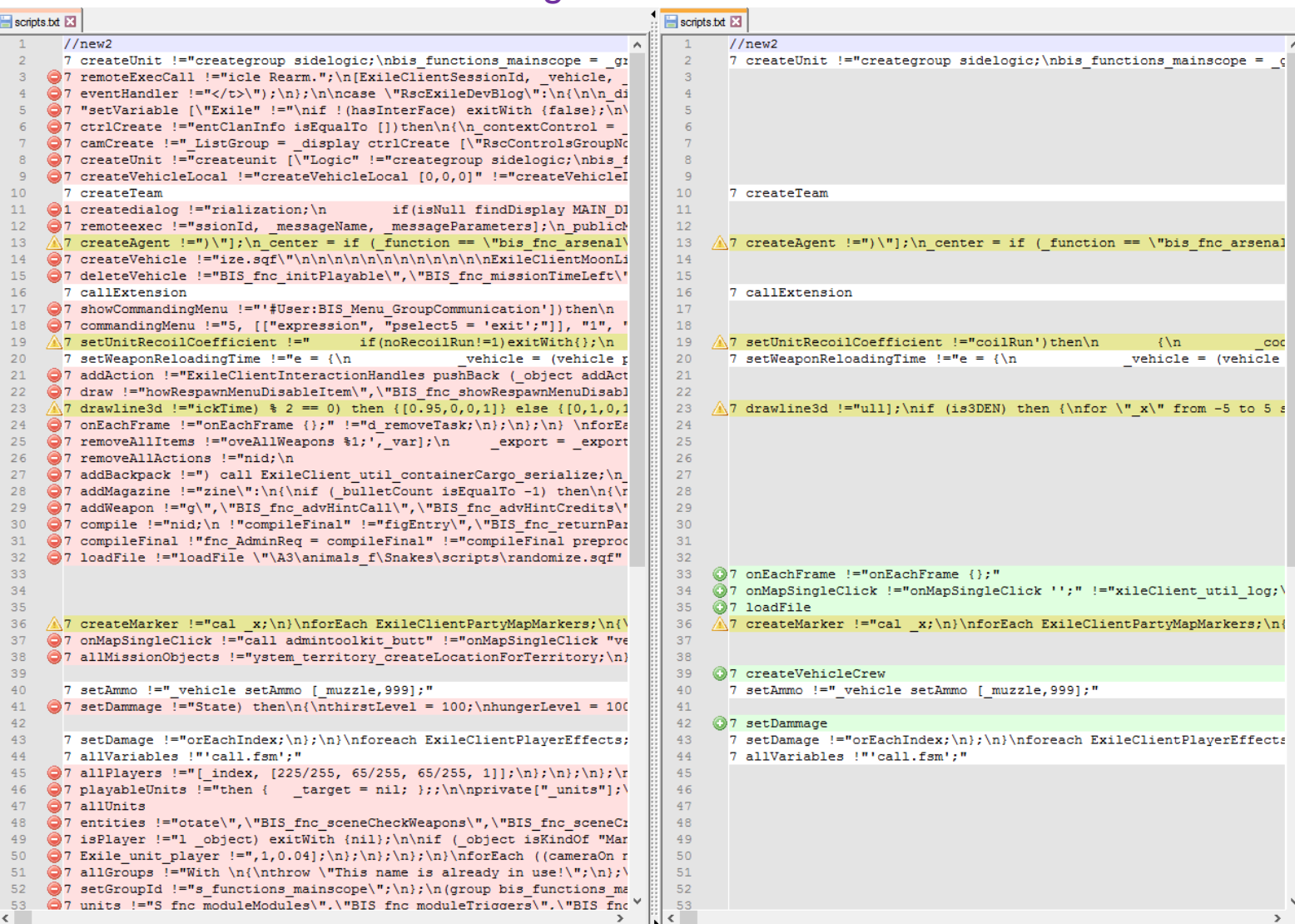Note: Make sure these are the only two files open in Notepad++

Final Step: Click on Plugins > Compare > Compare

You will get a window like this:

Be advised this method isn't perfect, however, it will get you most of the way there. The lines you want to pay close attention to are the **yellow** lines. These lines differ in a way so you have to find out what is different about them. The lines that are **red** are not on the other script.txt. Meaning if this is the script.txt from your server don't do anything. If this is the new script.txt you must add these lines to the other script.txt. The **white** lines mean that both script.txt matches. So you can just leave these alone. As for the **green** lines, these lines differ as well from the other document. You should check these lines as well to make sure there is nothing for you to add.

After you make your changes make sure to save
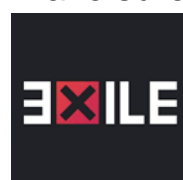
VIII. EDEN EDITOR

Note: Eden has a tooltip for what the mouse clicks are in bottom right of your screen.

Eden Editor is a tool that Bohemia gives us that has almost unlimited functionality when it comes to adding things or changing things within a map. We will cover several key functions within Eden. First though you need to understand the basics. **There will be several optional steps, these steps will be colored in blue.** Also, sometimes your map pbo is set for the old 2D editor there will be steps to convert the 2D maps into a version editable in Eden which is 3D.

Step 1: DePbo your map pbo (i.e. Exile.Altis).

Step 2: Place the folder created inside of your missions folder.
Note: This is a multipart step, **follow the pictures**. Also if you have more than one ARMA profile this may be slightly different look at the notes carefully.
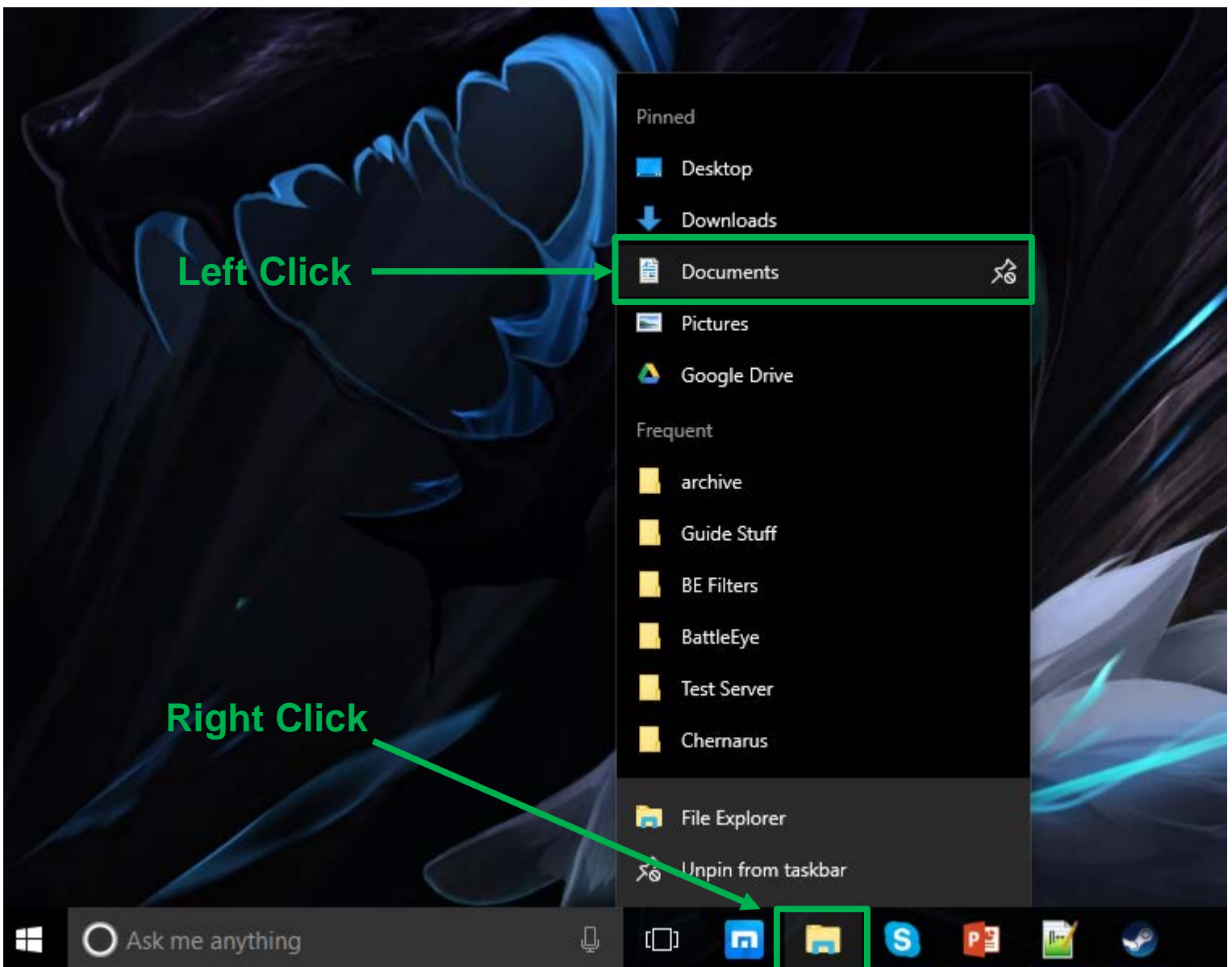
This next part is a bit tricky. You must determine if you have more than one ARMA profile. If you do have more than one you must determine which one you play on. Here's a picture from the ARMA profiles.
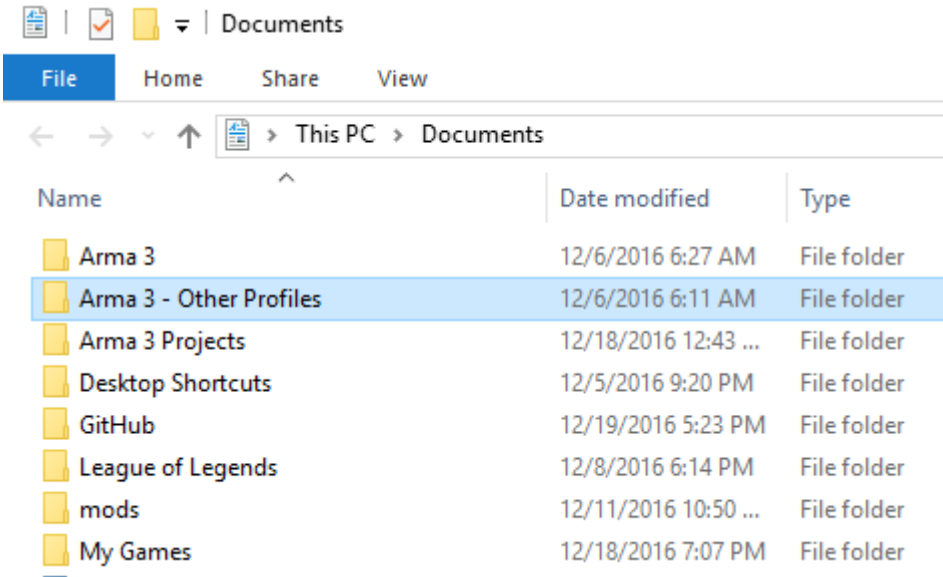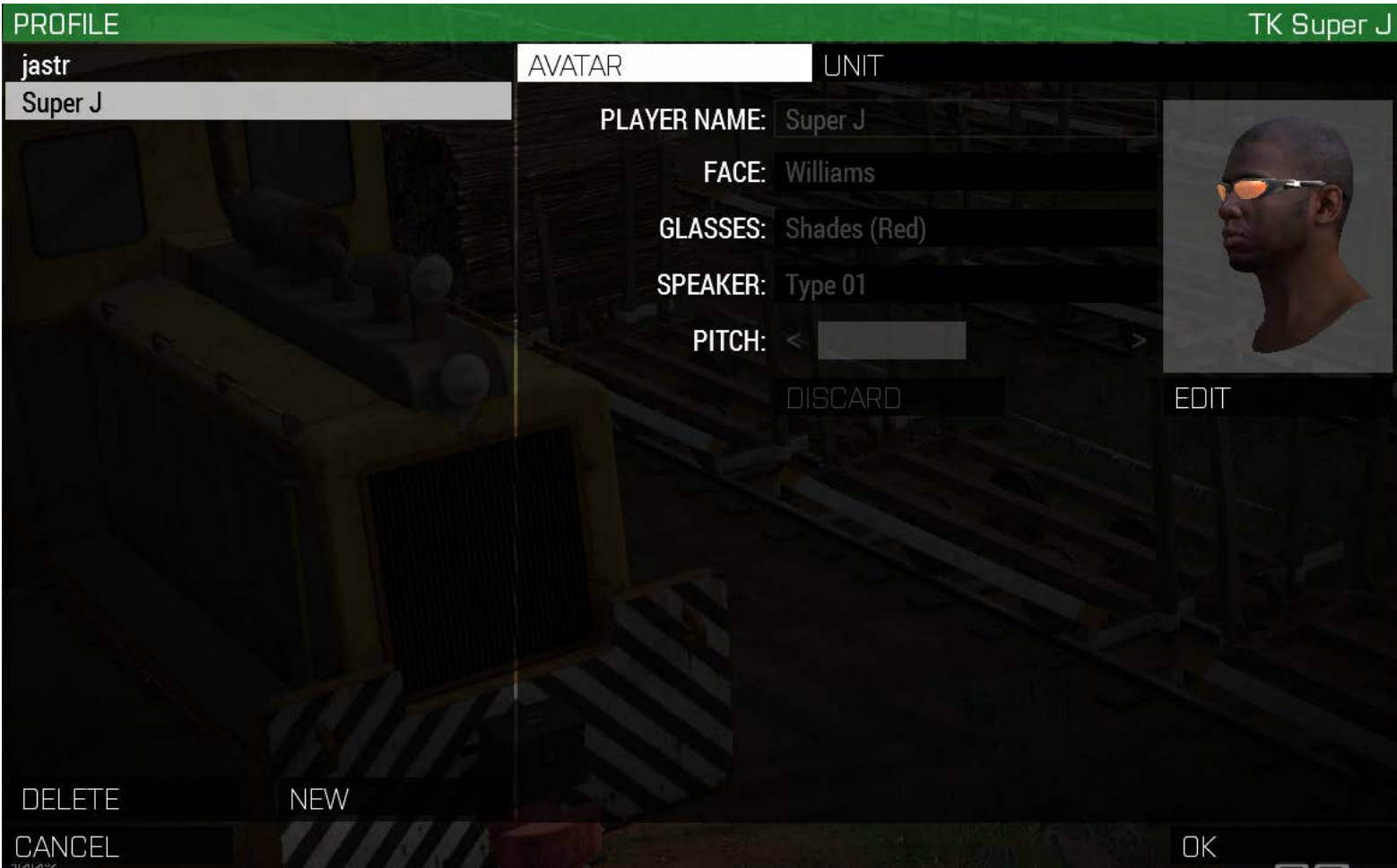
If you are like me and have more than one profile. You will go into the Other Profiles > missions folder. However, if you only have one profile you will go into the Arma 3 > Saved > missions folder. Once in missions folder you put your map folder into it.
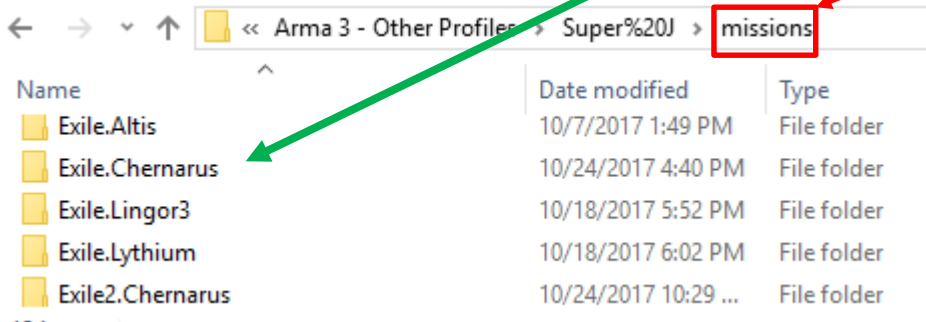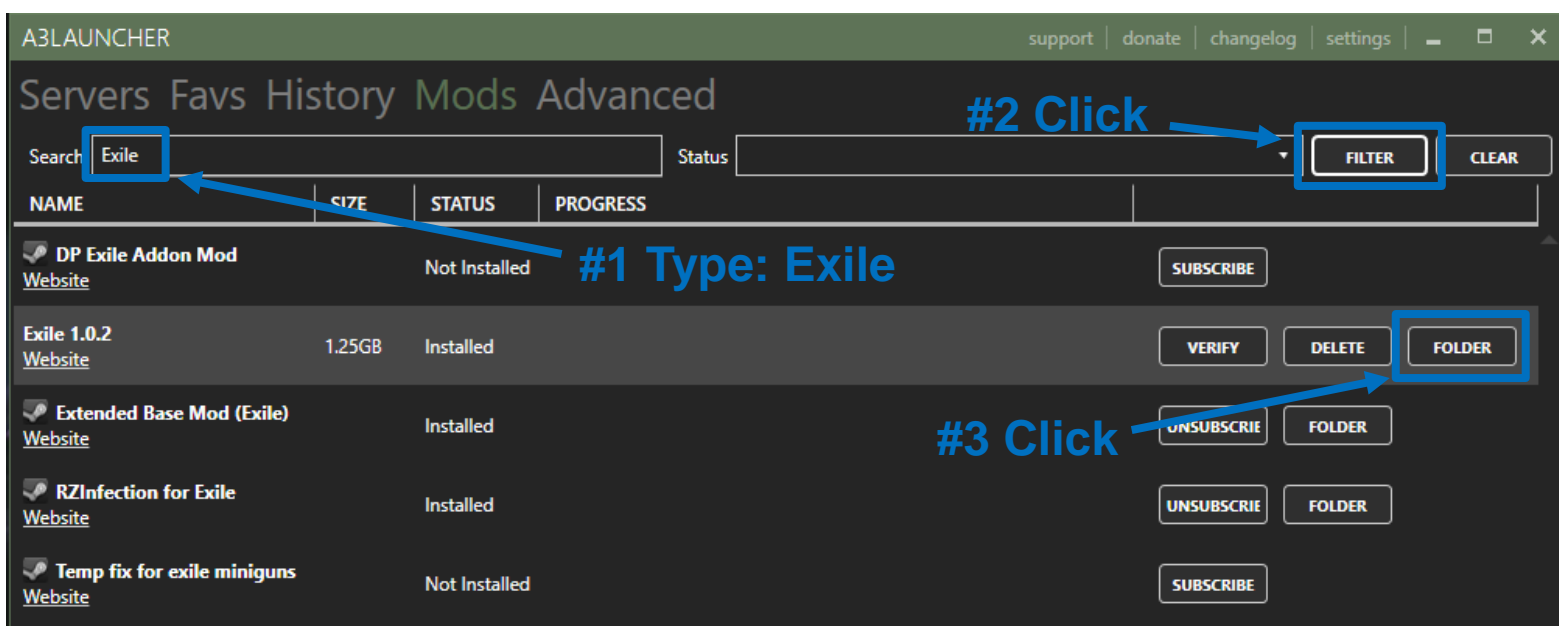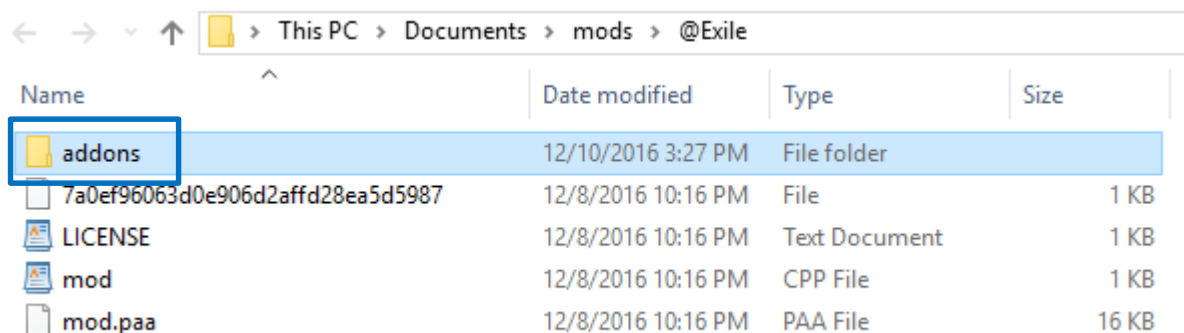
You may have to create this folder

Step 3: Download the 3DEN Plugin from Exile Mod website HERE.

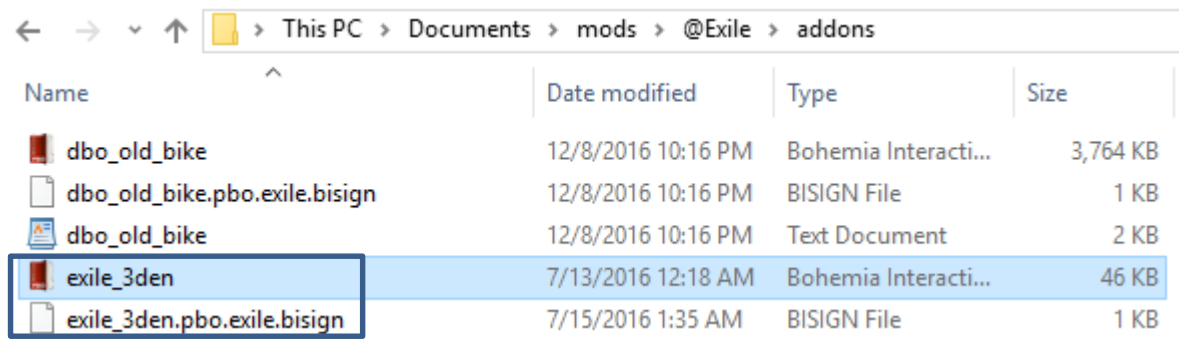Step 4: Install the 3DEN Plugin inside of you @Exile > addons folder.
Note: The @Exile folder may differ in location depending on which launcher you are using. For this example we will be using the A3Launcher. I recommend using this over normal launcher.

Step 5: With A3Launcher install the plugin M3Editor – 3DEN Edition.

These plugins are only necessary if you want to make it easier to add new things into your map. If I wanted to add more traders or a military base these plugins makes things easier to accomplish.

You may have to remove the exile_3den plugin before joining a server. If you get kicked from a server just remove it and add again it when you wish to edit.

Step 6: Launch your ARMA with Exile and M3Editor – 3DEN Edition (optional).
Note: Be careful what additional addons you load here. Below I have CBA and some CUP addons.
If I were to add any of these assets to my map in Eden then my server would also have to run these addons. Make sure this is what you want.

Step 7: Open Eden then open the map you wish to edit
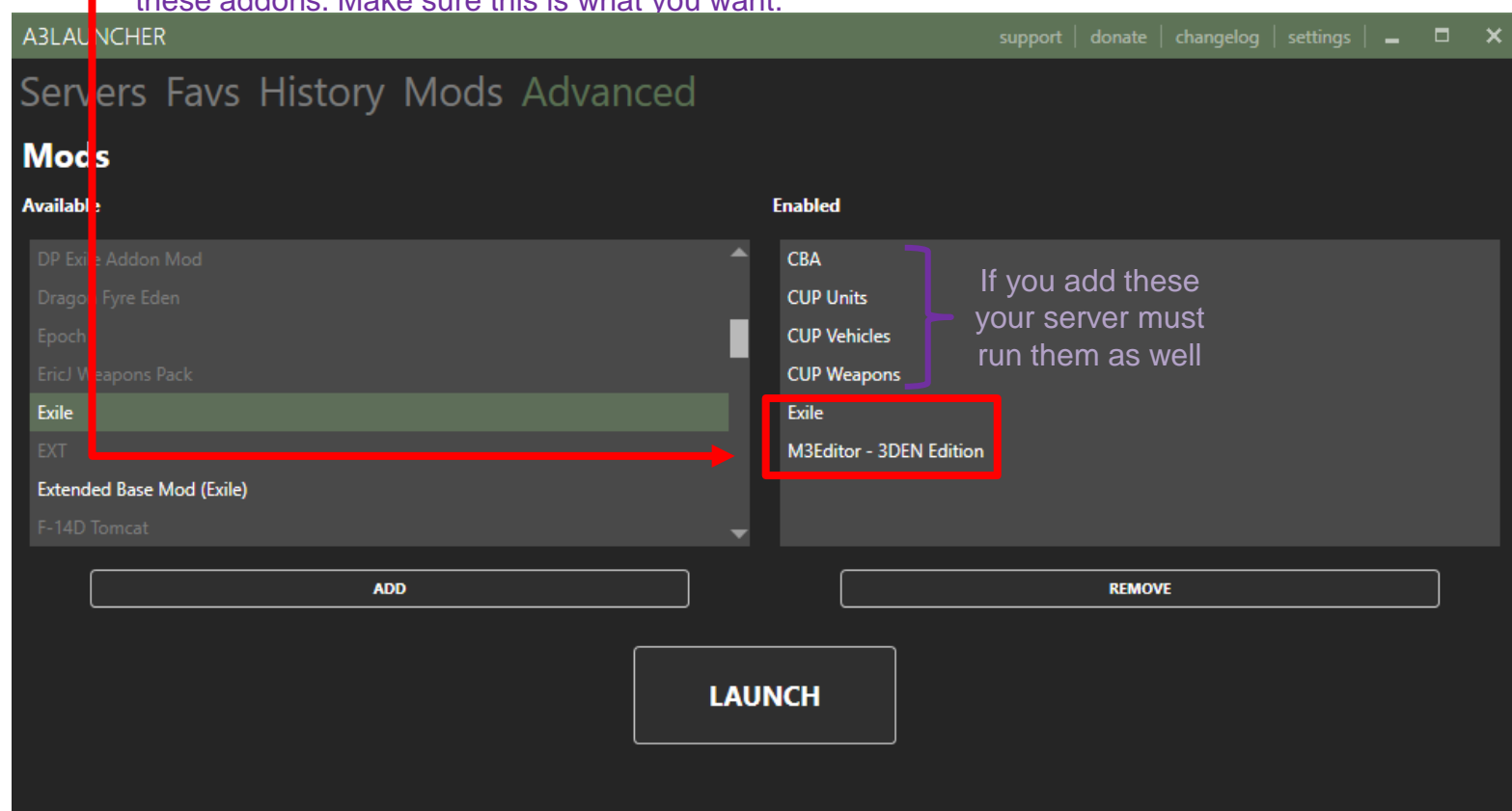
Step 8: Open your map from the mpmissions folder
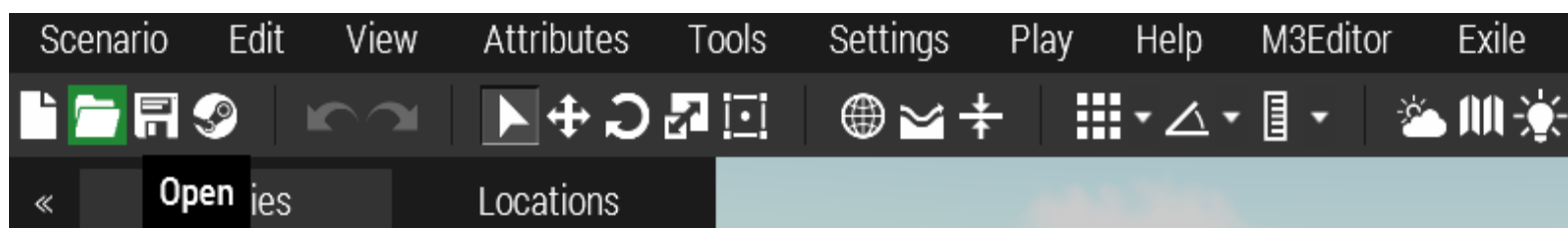
Whether it says import or open just click it.

BOHEMIA MADE
AND UPDATE TO
EDEN. THIS
WORKS NOW.
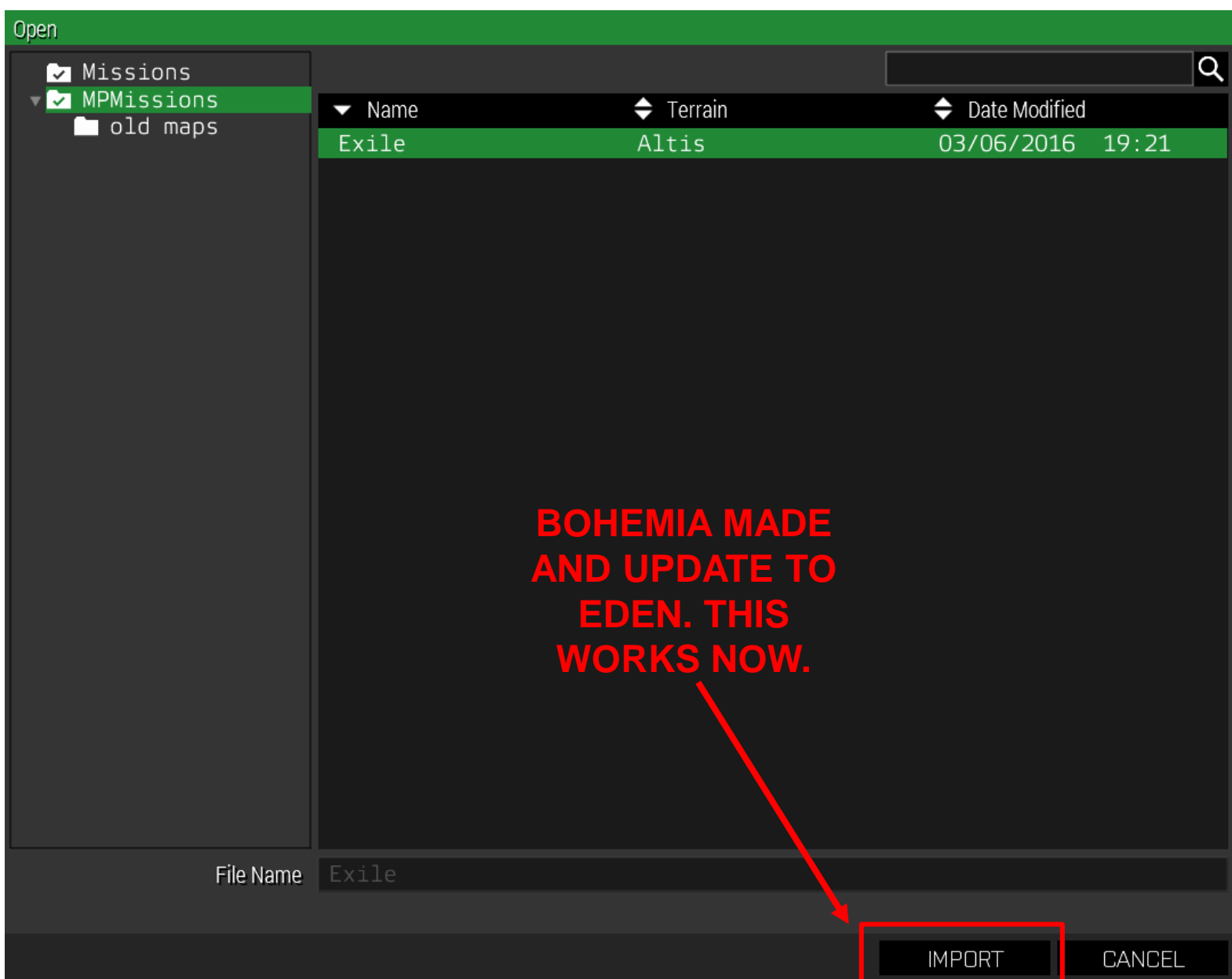
Step 9: Click on multiplayer and then server browser.

1. **<u>Creating a new spawn zone</u>**: This section will be covering making a new spawn zone. In order to do this you must currently already have spawn zones on your map. Also, Eden is just like any other Windows application. Features like copy (Ctrl+C), paste (Ctrl+V), and pressing the delete key also works within Eden. I 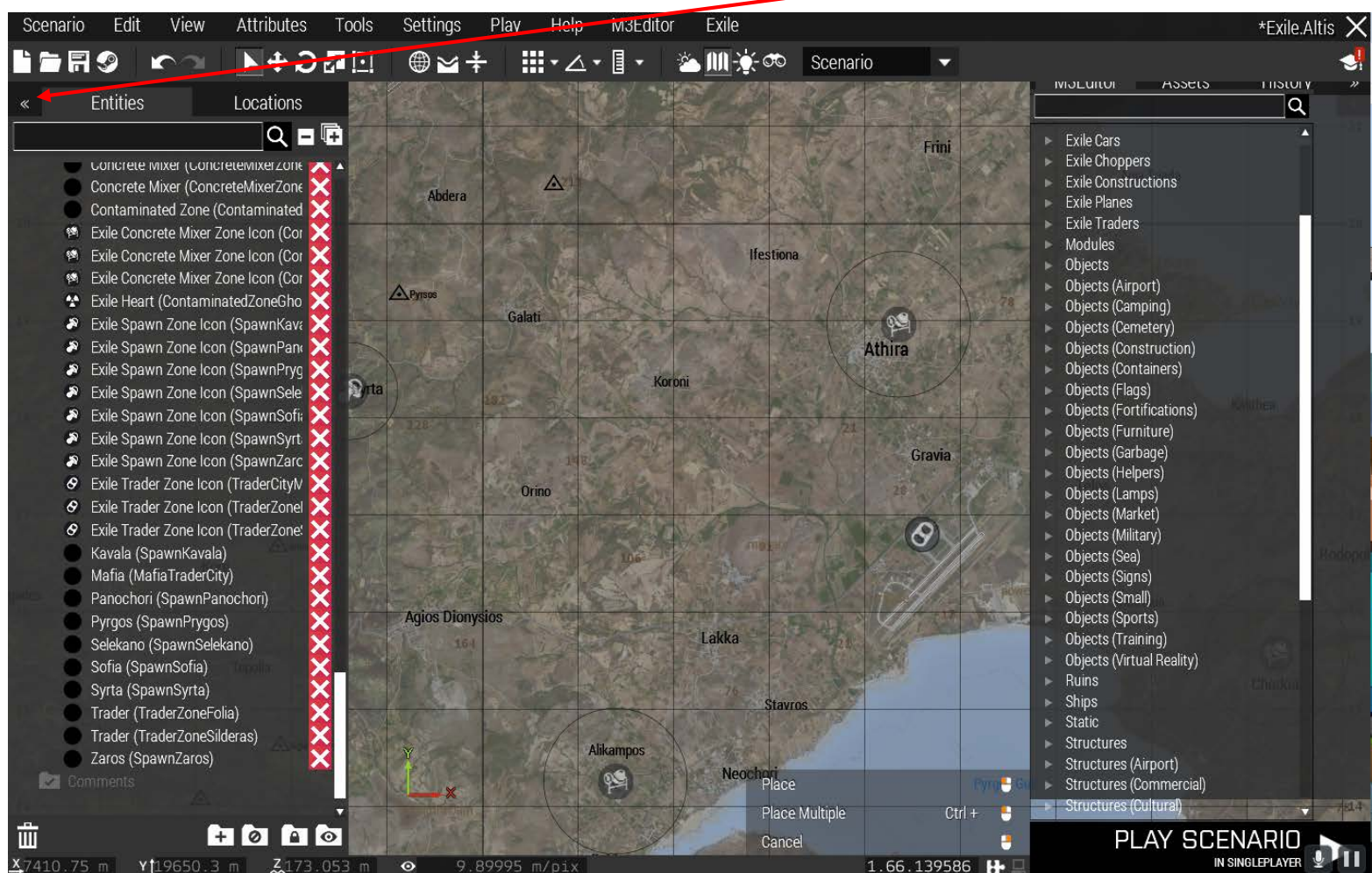will only tell you to copy, paste, or delete. Lastly, where you paste something at will happen at wherever your mouse is at the moment you press Ctrl+V. You can, however, drag things around though.

Step 1: Open the side bar for your entities by clicking HERE.

## For this example we will be creating a new spawn at Orino

Side Note: When it is time RePbo your map make sure that more than just mission.sqm is present. If all you see is the mission.sqm after you save just copy the mission.sqm into a map pbo that has all of your files in it. If you had to import, you will know you have done everything correctly when you see the mission.sqm.oldBackup file after you save.

Step 2: Copy any spawn and paste it on the place you desire.
Drag around if necessary.
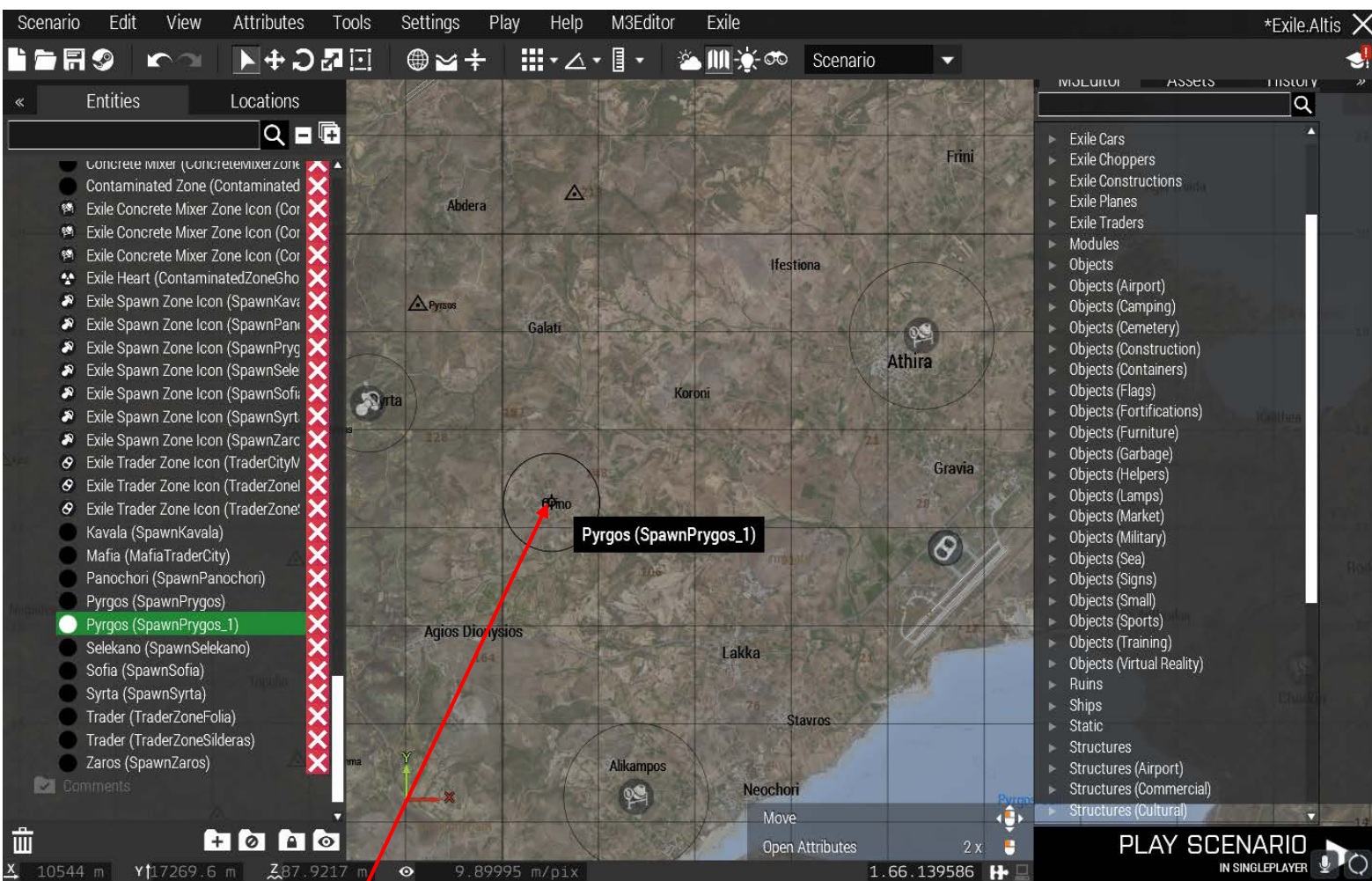
Step 3: Double click on the new position to open the attributes.

You will do Step 2 with the Spawn Zone Icon in Step 5.

Side Note: From the File you can click on Show Required Addons. From this you can see what addons that your server **MUST** have installed in order to run your map. In most cases these addons will go in the clientside mod category. The only exception to this is if you see 3DEN here. You can run 3DEN serverside.

Part of
CUP_Vehicles

Part of CUP_Units
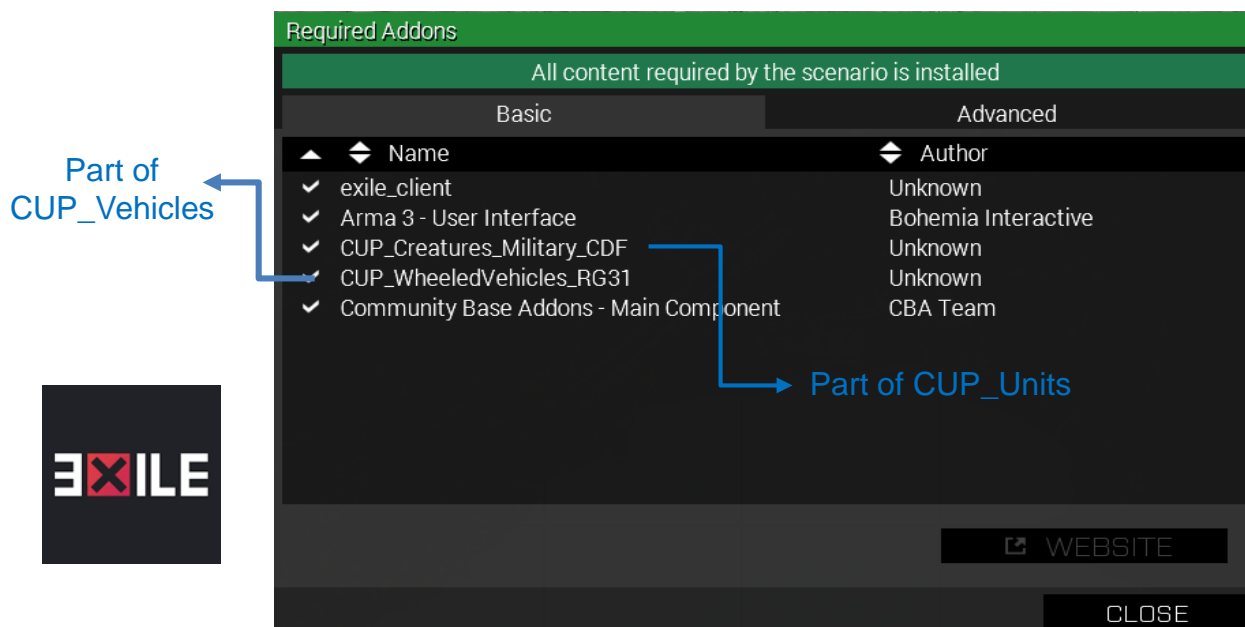
Step 4: Rename your Variable Name ensuring that Spawn is the first word. Also, rename the Text to the same thing. Adjust the size of your spawn area as well. This is the radius in which players will spawn in. Note: Do not forget your position you will need this on next step.

Edit: Pyrgos (SpawnPrygos_1)

Marker: Init

Variable Name | SpawnOrino
Text | Orino

Marker: Transformation

Position | X 10514.328 | Y 17259.748
Spawn Radius → Size | A 550 | B 550
Rotation | Z 0

Marker: Style

Shape

Brush | Border
Color | Black
Alpha | ◄ | ► 100%

Marker: Exile

Zone
Icon

Click OK when finished

OK   CANCEL

If you don't like the size of your spawn just come back into this and edit it again. The black lines show you the radius.

EXILE   EXILE

Step 5: After creating your icon from Step 3. Rename your spawn zone icon the same as the spawn zone ensuring Icon is the last word. Make sure the positions match each other.

Edit: Exile Spawn Zone Icon (SpawnSofiaIcon_1)

▶ Marker: Type

▸ Marker: Init

Variable Name | SpawnOrinoIcon
Text |

▸ Marker: Transformation

Position | X | 10514.328 | Y | 17259.748
Size | A | 1 | B | 1
Rotation | Z | 0

▸ Marker: Style

Color | Default
Alpha | ◀ | | ▶ | 100%

▸ Marker: Exile

Zone | ▼
Icon | ▼

OK   CANCEL

2. **Creating new traders**: One thing to understand when creating new traders is first where they are located at in your map pbo. Inside of your map pbo (i.e. Exile.Altis) is a file called initPlayerLocal.sqf. Traders are the one thing that when created in Eden **MUST** be exported in order to load properly. To do this you will need the two 3DEN plugins from . Lastly, when creating new traders you must have a map pbo that currently has traders in order to obtain the zones.

Step 1: Open any pbo that contains traders (does not have to be the one you wish to edit). Once open go to entities right click any Trader Zone Icon and click Save Custom Composition.
Note: You will repeat this step for the Trader Zone as well.
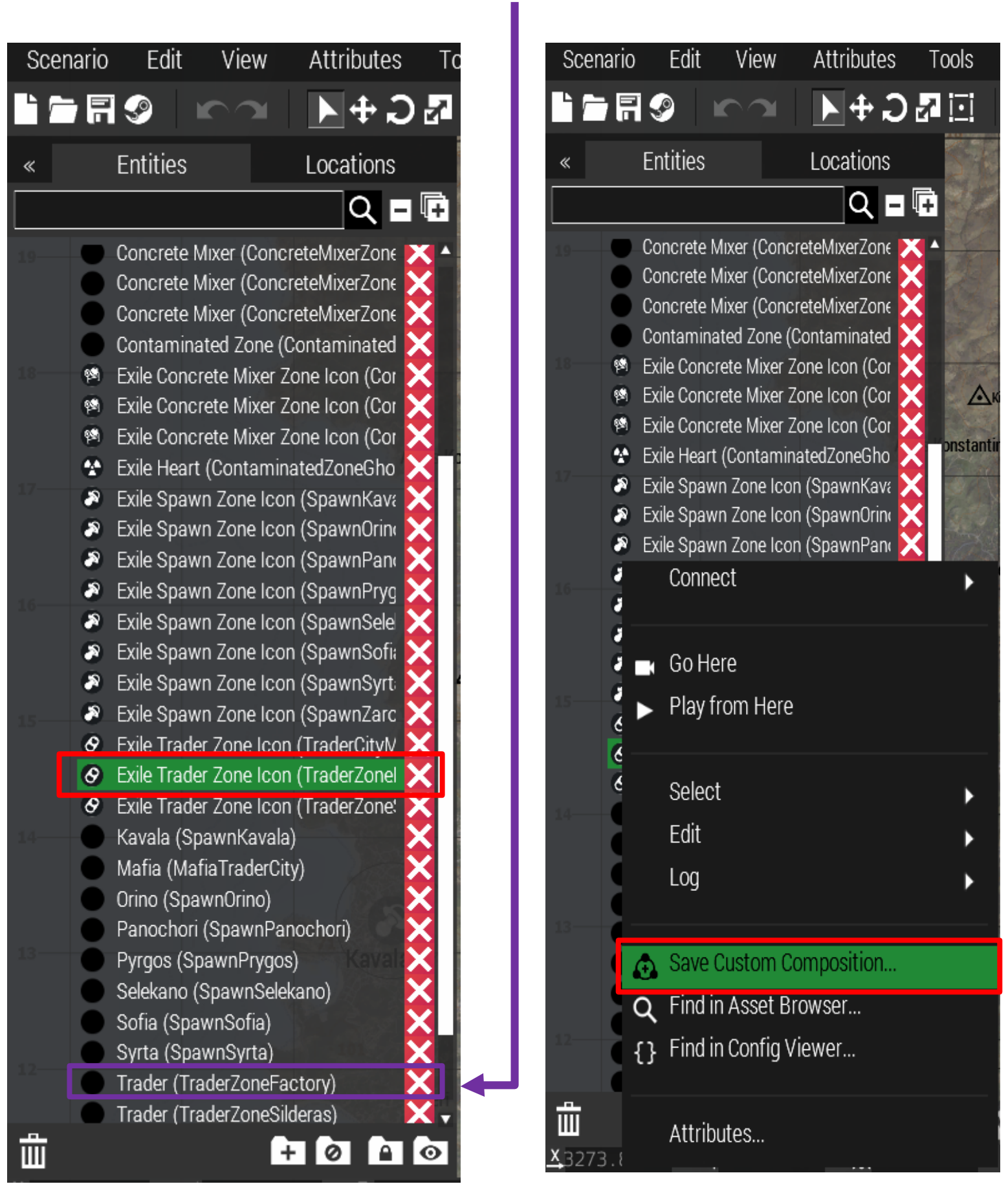
Step 2: Name your new composition TraderIcon and save it inside of the Other category.
Note: Once you create these they remain within your Eden permanently.

Save Custom Composition

<New Composition>
▼ Other
    TraderIcon
    TraderZone
    SpawnZone

Title
TraderIcon

Author
Super J

Category
Other ▼

OK   CANCEL

M3Editor   Assets   History   »
F1   F2   F3   F4   F5   F6

Take note of the tabs
where the new custom
compositions are so that
you know how to access
them when you need them.

▼ Other
    TraderIcon
    TraderZone
    SpawnZone

Step 3: Place your new Trader Zone and Trader Zone Icon compositions on the map where you want them. Ensure that you select the drop down box Safe Zone and Trader City for your Trader Zone. You do not have to have these options selected for the Trader Zone Icon.

Edit: Trader (TraderZoneFactory)

◢ Marker: Init

Variable Name | TraderZoneFactory
Text | Trader

◢ Marker: Transformation

Position | X | 6151.656 | Y | 16246.557
Size | A | 175 | B | 175
Rotation | Z | 0

◢ Marker: Style

Shape

Brush | Border
Color | Black
Alpha | ◄ | 100%

◢ Marker: Exile

Don't forget to select these

Zone | Safe Zone
Icon | Trader City

OK   CANCEL

Remember to make sure the positions match for both the Icon and the Zone. You can drag them around if necessary.

Step 4: Get your camera in position to start adding traders then press M. Start adding props and furniture using M3Editor.
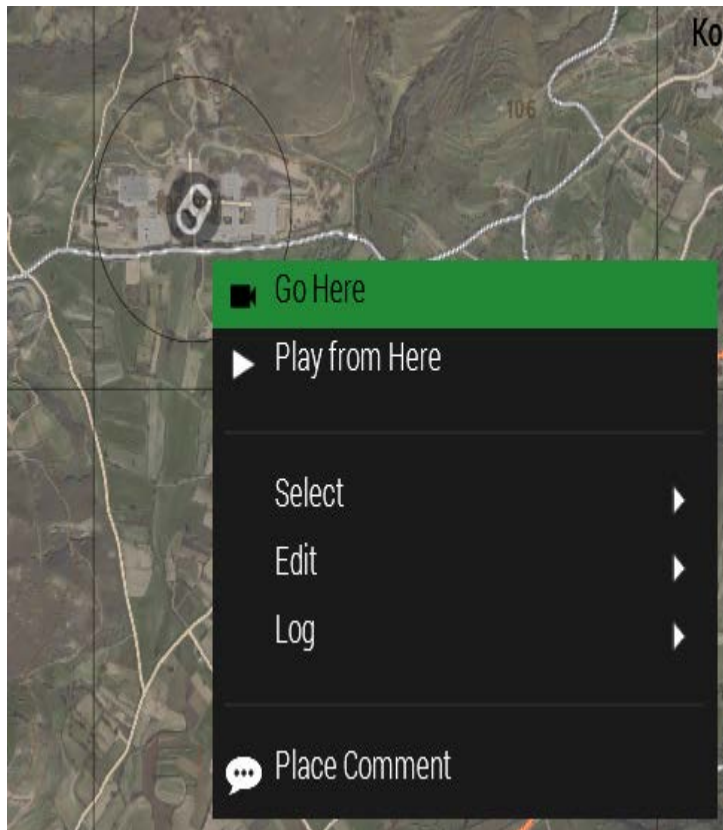
Moving your camera in Eden is just like walking in game. You can use the same controls to move you around. Also M3Editor is the one function in Eden that you must double click to add. Once you get the item down you can drag it around.
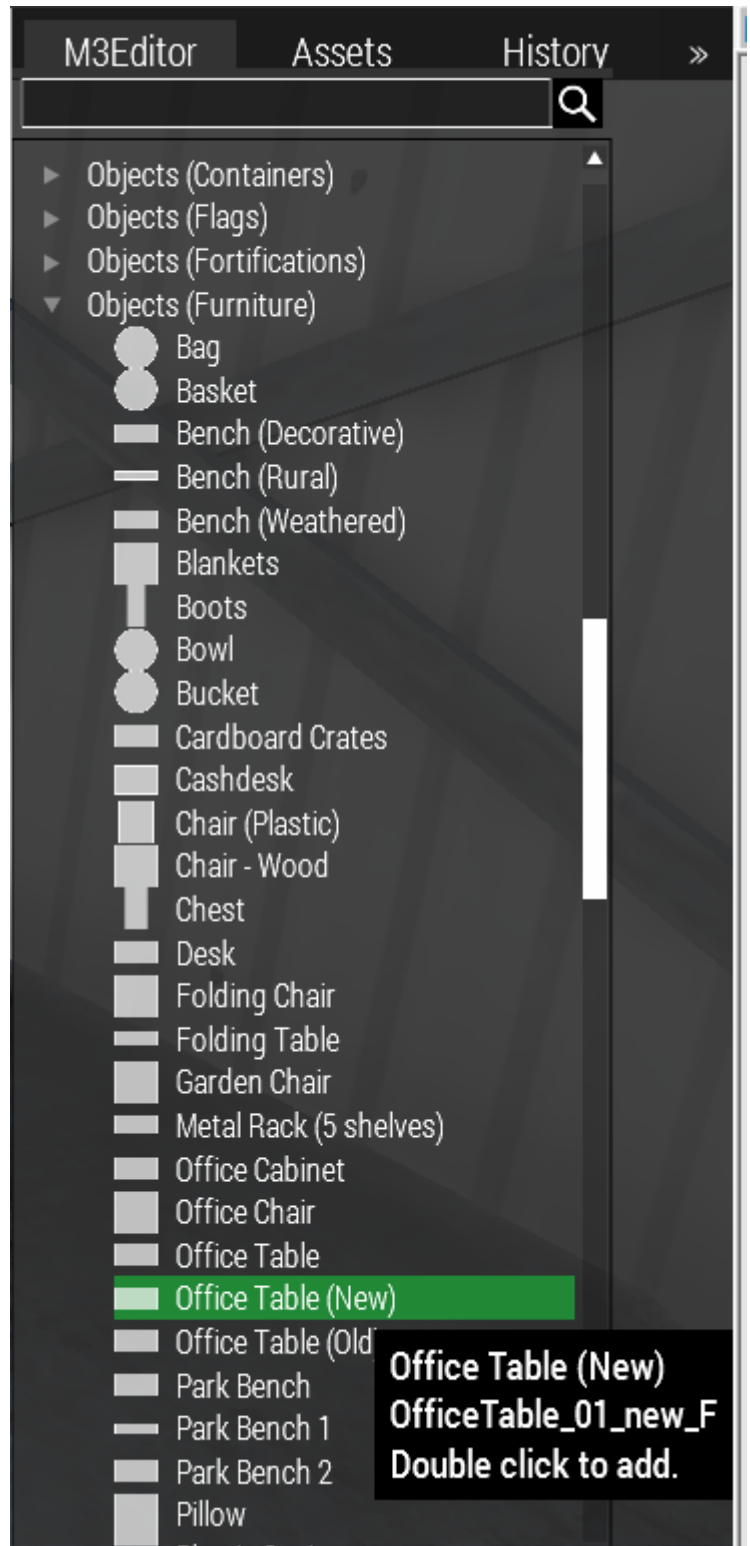
Edit: Exile Armory Trader

▶ Object: Type

▶ Object: Init

◢ Object: Transformation

Position	X	6143.198	Y	16203.604	Z	0
Rotation	X	0	Y	0	Z	0

Placement Radius 0

▶ Object: Control

▶ Object: States

▶ Object: Special States

◢ Object: Identity

Name Wallace Thompson

Face Campbell

Call Sign No Call Sign

Voice American English 01

Voice Pitch 0.99x

Insignia None

▶ Object: Presence

◢ Object Specific - Object: Exile Armory Trader

Trader Type Exile_Trader_Armory

Animations InBaseMoves_SittingRifle1

OK   CANCEL

You can set custom animations for your traders. Here is a few:

InBaseMoves_SittingRifle1
InBaseMoves_table1
InBaseMoves_sitHighUp1
InBaseMoves_Lean1
InBaseMoves_repairVehicleKnl

InBaseMoves_repairVehiclePne
HubStandingUA_move1
InBaseMoves_patrolling2
InBaseMoves_patrolling1
HubStandingUA_idle1

If you don't like the animation you can simply change or delete by coming back in here.

Step 6: From the Exile tab click Save initPlayerLocal.sqf and then click Copy. Save your mission in Eden and then close Eden.

Doing this saves your traders to your clipboard so that you can then save them to the initPlayerLocal.sqf inside of your map pbo.

If you don't have 3DEN you won't be able to do this.

Exile

Run Sanity Check

Convert into Simple Object

Synchronize Simple Object   0 [NUM]

🖫 Save initServer.sqf

🖫 Save initPlayerLocal.sqf

🖫 Save Markers and Zones

🖫 Save Weather Keyframe

initPlayerLocal.sqf

```
/**
 * Created with Exile Mod 3DEN Plugin
 * www.exilemod.com
 */

if (!hasInterface || isServer) exitWith {};

// 2 NPCs
private _npcs = [
["Exile_Trader_Armory", ["InBaseMoves_SittingRifle1"],
"Exile_Trader_Armory", "WhiteHead_11",
[["srifle_DMR_06_olive_F","","","",[],[],""],[],[],["U_Rangemaster"
,[]],["V_Rangemaster_belt",[]],[],"H_Cap_headphones","G_Shades_Blac
k",[],["","","","","",""]], [14568.1, 16764.3, 18.0364], [0, 1, 0],
[0, 0, 1]],
["Exile_Trader_Armory", ["InBaseMoves_SittingRifle1"],
"Exile_Trader_Armory", "WhiteHead_05",
[["srifle_DMR_06_olive_F","","","",[],[],""],[],[],["U_Rangemaster"
,[]],["V_Rangemaster_belt",[]],[],"H_Cap_headphones","G_Shades_Blac
k",[],["","","","","",""]], [6143.18, 16202.7, 43.3723], [0, 1, 0],
[0, 0, 1]]
];

{
    private _logic = "Logic" createVehicleLocal [0, 0, 0];
    private _trader = (_x select 0) createVehicleLocal [0, 0, 0];
    private _animations = _x select 1;

    _logic setPosWorld (_x select 5);
    _logic setVectorDirAndUp [_x select 6, _x select 7];

    _trader setVariable ["BIS_enableRandomization", false];
    _trader setVariable ["BIS_fnc_animalBehaviour_disable", true];
```
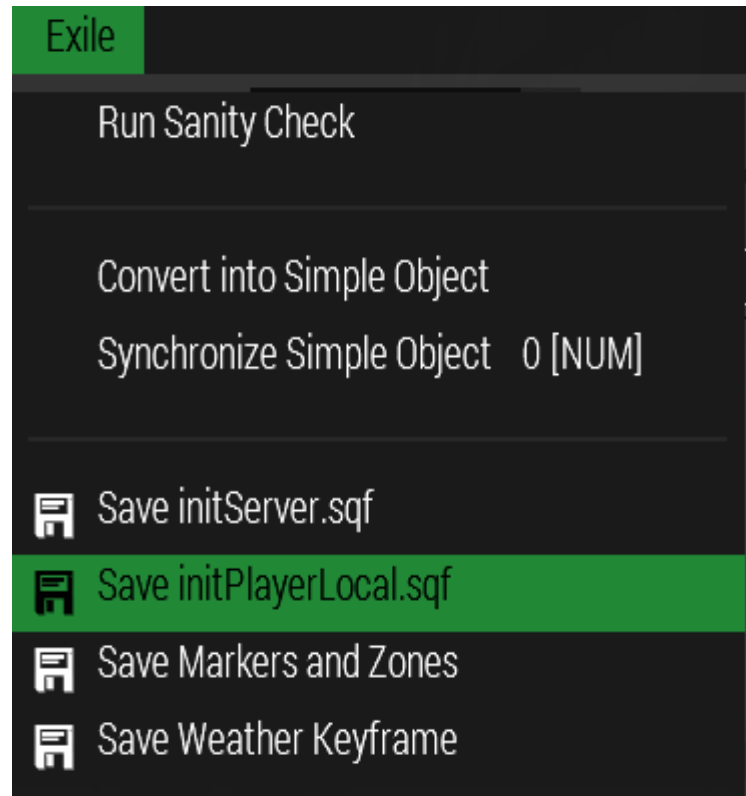
COPY   CLOSE

Final Step: Paste your new traders inside of your initPlayerLocal.sqf using Dash and save. Once complete RePbo your map pbo and upload back to your server.

File   Edit   Project   Compile   Help

description.ext   exile.ini   tutorial.txt   initPlayerLocal.sqf

```
 1  /**
 2   * Created with Exile Mod 3DEN Plugin
 3   * www.exilemod.com
 4   */
 5
 6  if (!hasInterface || isServer) exitWith {};
 7
 8  // 2 NPCs
 9  private _npcs = [
10  ["Exile_Trader_Armory", ["InBaseMoves_SittingRifle1"], "Exile_Trader_Armory", "WhiteHead_11", [["srifle_DMR_06_olive_F","","","
11  ["Exile_Trader_Armory", ["InBaseMoves_SittingRifle1"], "Exile_Trader_Armory", "WhiteHead_05", [["srifle_DMR_06_olive_F","","","
12  ];
13
14  {
15      private _logic = "Logic" createVehicleLocal [0, 0, 0];
16      private _trader = (_x select 0) createVehicleLocal [0, 0, 0];
17      private _animations = _x select 1;
18
19      _logic setPosWorld (_x select 5);
20      _logic setVectorDirAndUp [_x select 6, _x select 7];
21
22      _trader setVariable ["BIS_enableRandomization", false];
23      _trader setVariable ["BIS_fnc_animalBehaviour_disable", true];
24      _trader setVariable ["ExileAnimations", _animations];
25      _trader setVariable ["ExileTraderType", _x select 2];
26      _trader disableAI "ANIM";
27      _trader disableAI "MOVE";
28      _trader disableAI "FSM";
29      _trader disableAI "AUTOTARGET";
30      _trader disableAI "TARGET";
31      _trader disableAI "CHECKVISIBLE";
32      _trader allowDamage false;
33      _trader setFace (_x select 3);
34      _trader setUnitLoadOut (_x select 4);
35      _trader setPosWorld (_x select 5);
36      _trader setVectorDirAndUp [_x select 6, _x select 7];
37      _trader reveal _logic;
38      _trader attachTo [_logic, [0, 0, 0]];
39      _trader switchMove (_animations select 0);
40      _trader addEventHandler ["AnimDone", {_this call ExileClient_object_trader_event_onAnimationDone}];
41  }
42  forEach _npcs;
```

IX. CREATING DMS AI STATIC MISSION

Links: Sample_Mission, Sample_Objects

In this section we will be covering how to create your own static AI mission using Defents Mission System. To get you started I will provide you a template that I use for my own missions. In this section we will cover only land as sea missions are more difficult. I will not be covering the basic things such as DePbo, RePbo, and Eden Editor basics. If you do not know how to do these things go back and read them in the guide. At this point in the guide I am assuming you have a general knowledge of scripting and using Eden.

Note: You want to run ARMA 3 in full screen window mode as you will be doing a lot of going back and forth. It helps to have two monitors as well.

*Any steps in **BLUE** are optional and depend on your desires for your mission.

Step 1: Open the Sample_Mission using either Dash or Notepad++.

Step 2: Ensuring you have 3Den installed, open Eden Editor and load your map.
Note: We are opening a blank map and not your actual pbo.

Step 3: Place your objects down that will create your AI Base.

Step 4: Place your AI ensuring that they are semi suspended in the air.
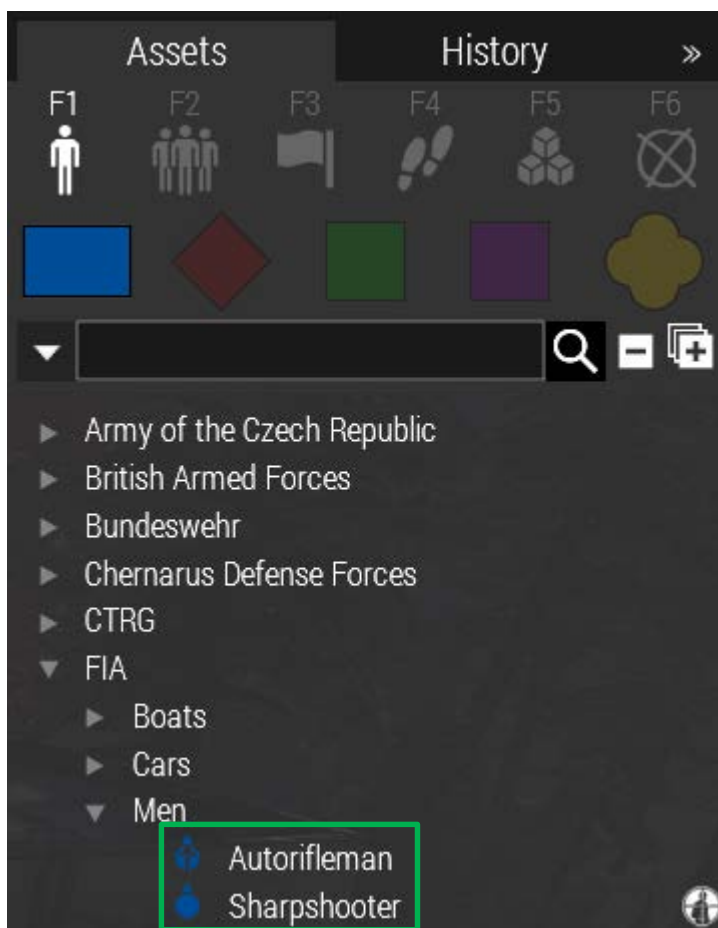
The type of unit doesn't matter. We just want to see where our guy fits at in the mission.

When you place him, adjust his Z position so that his feet are just barely off the surface. Don't want our guy spawning in an object.

Object: Transformation

| Position | X | 5154.22 | Y | 2374.39 | Z | 10.9 |
| Rotation | X | 0 | Y | 0 | Z | 303.147 |

Placement Radius 0

```
private _AIGroup =
[
    [
        [5154.22,2374.39,10.9],
        [13164.80,10670.60,0.926],
        [13185.19,10713.01,1.852],
        [13175.48,10709.06,1.238],
        [13194.20,10719.34,2.125],
        [13186.01,10738.35,1.085],
        [13185.26,10680.34,8.653],
        [13179.64,10697.44,0.767],
        [13197.35,10712.51,8.824],
        [13119.43,10761.06,0.973]
    ],
    10,
    "hardcore",
    "custom",
    _side,
    _AIGearSet
] call DMS_fnc_SpawnAIGroup_MultiPos;
```

This is your AI count. Ensure that you count the AI positions and change this number to the amount of positions you have.

Step 6: Place a single AI in the center in which to spawn an AI group. Put the position in _SniperGroup.

If you want to change the type of AI you may do this here. You don't have to change the name of the group just this.

If you change this here you will have to change it in the rest of the script. Just leave it alone.

```
private _SniperGroup =
[
    [13137.7,10329.7,0],
    5,
    "hardcore",
    "sniper",
    "bandit"
] call DMS_fnc_SpawnAIGroup;
```

DMS has multiple types of AI to spawn they are: sniper, assault, MG, and random.

Step 7: Equip your AI with weapons, ammo, backpack, etc.

```
private _AIGearSet =
[
    "CUP_lmg_L110A1", // "Weapon"
    [],
    [["CUP_200Rnd_TE4_Red_Tracer_556x45_L110A1",3]], // "Ammo","clips"
    "",
    [],
    ["Rangefinder","ItemGPS","NVGoggles"],
    "",
    "H_HelmetSpecO_blk", // "Helmet"
    "U_O_SpecopsUniform_blk", // "Uniform"
    "V_PlateCarrierSpec_blk", // "Vest"
    "B_TacticalPack_blk"   // "Backpack"
];
```

Tip: Pull class names out of your traders information. That way you don't have to go digging to find what you want.

ƎXILE
ƎXILE

Step 8: Place down your static machine guns and place them in _staticGuns.

Turrets
- C-RAM
- Mk-29 GMLS
- Mk-49 GMLS
- Mk21 Centurion
- Mk30 HMG .50
- **Mk30 HMG .50 (Raised)**

```
private _staticGuns =
[
    [
        [13147.86,10631.03,1.098],
        [13160.77,10627.63,1.094],
        [13163.74,10746.43,6.251]
    ],
    _AIGroup,
    "assault",
    _difficulty,
    "bandit",
    "random"
] call DMS_fnc_SpawnAIStaticMG;
```

Step 9: Place your mission reward crate down and copy the position over then fill the crate.

supply

Supplies
- Ammo
  - Supply Box
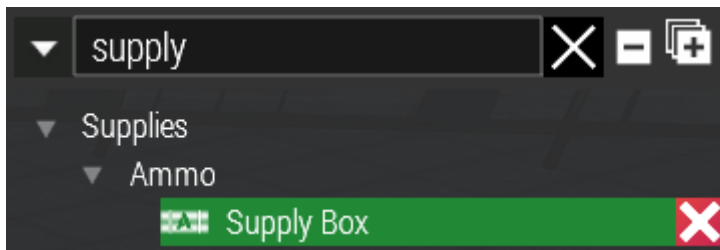
```
private _crate = ["Exile_Container_SupplyBox",[0,0,0]] call DMS_fnc_SpawnCrate;
_crate setPosATL [13165.97,10702.73,2.673]; // *Change This*
_crate setVariable ["DMS_AllowSmoke", false];
[_crate, "SuperGear"] call DMS_fnc_FillCrate;


DMS_CrateCase_SuperGear =
[
    [
        "CUP_launch_Igla",
        "CUP_launch_Igla",
        "CUP_launch_Javelin",
        "CUP_launch_Javelin",
        "CUP_launch_FIM92Stinger",
        "CUP_launch_FIM92Stinger",
        "CUP_launch_NLAW",
        "CUP_launch_NLAW",
        "CUP_launch_MAAWS"
    ],
```

This is the first block with weapons. The second block below this is ammo and last block is backpacks. You can place other items as well in last block.

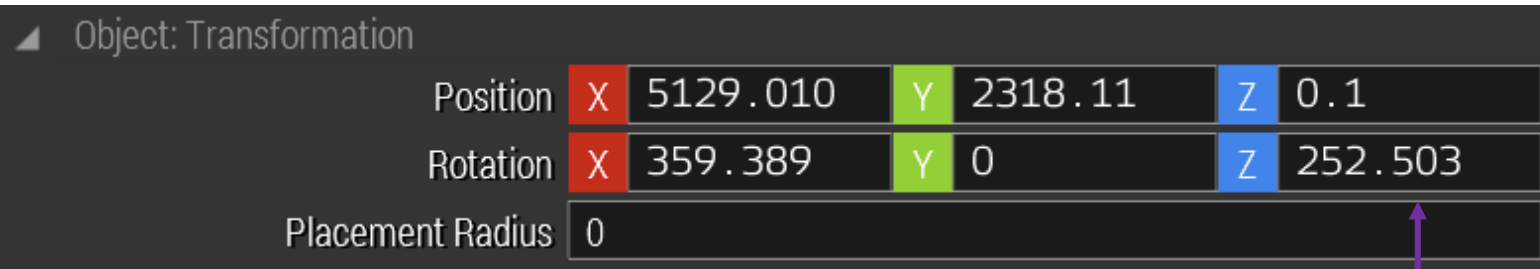Tip: As stated earlier, pull class names out of your traders information.

Step 10: Place down the vehicles you want as a mission reward then add them to the script.
Note: This is a multi part step follow the pictures below. Same applies to the helicopter.

LAV	✕ ▬ ⊡

▼ United States Marine Corps
   ▼ APCs
      🧍 LAV-25A1

After your vehicle snaps into place, manually raise it by 0.1 so that it barely sits above the surface. Otherwise you risk your vehicle being damaged after spawning in. For instance if after you snap it in the Z is at 0 change it to 0.1

◢ Object: Transformation

	X	5129.010	Y	2318.11	Z	0.1
Position	X	5129.010	Y	2318.11	Z	0.1
Rotation	X	359.389	Y	0	Z	252.503

Placement Radius | 0

```
private _truckl = ["CUP_B_LAV25_USMC",[0,0,0]] call DMS_fnc_SpawnNonPersistentVehicle;
_truckl setDir (252.503);
_truckl setPosATL [5129.01,2318.11,0.1];
_truckl setVariable ["DMS_AllowSmoke", false];
```

This is the direction the vehicle is facing. Not to be confused with the height.

🔾 Save Custom Composition...
🔍 Find in Asset Browser...
{} Find in Config Viewer...
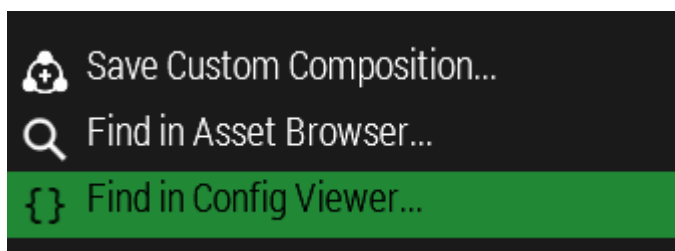
This is your vehicle's class name.

Path: configfile >> "CfgVehicles" >> "CUP_B_LAV25_USMC"

Step 11: Change the mission name and messages to whatever you want.

```
// Define Mission Start message
_msgStart = ['#FFFF00', "What you want to say when you mission spawns"]; // *Change This*

// Define Mission Win message
_msgWIN = ['#0080ff',"What you want to say when a player completes the mission"]; // *Change This*

// Define Mission Lose message
_msgLOSE = ['#FF0000',"What you want to say if the mission is not completed in time"]; // *Change This*

// Define mission name (for map marker and logging)
_missionName = "Sample Mission"; // *Change This*
```

Ǝ✗ILE   **SAVE MISSION NOW!**   Ǝ✗ILE

Exile

Run Sanity Check

Convert into Simple Object

Synchronize Simple Object   0 [NUM]

Save initServer.sqf

Place the Sample_Object.sqf inside of the objects > static folder within DMS.

```
initServer.sqf
/**
 * Created with Exile Mod 3DEN Plugin
 * www.exilemod.com
 */

ExileRouletteChairs = [];
ExileRouletteChairPositions = [];

// 4 Vehicles
private _vehicles = [
["Land_Cargo_Tower_V1_No1_F", [5076.05, 2294.21, 21.8862],
[0.781083, -0.624428, 0], [0, 0, 1], true],
["Land_Cargo_Tower_V1_No4_F", [5097.81, 2276.47, 21.8411],
[0.754176, -0.656673, 0], [0, 0, 1], true],
["Land_Cargo_Tower_V1_No7_F", [5092.79, 2308.86, 21.8862],
[0.739385, -0.673282, 0], [0, 0, 1], true],
["Land_Cargo_Tower_V1_No7_F", [5110.47, 2289.71, 21.8446],
[0.720386, -0.693573, 0], [0, 0, 1], true]
];

{
    private _vehicle = (_x select 0) createVehicle (_x select 1);
    _vehicle allowDamage false;
    _vehicle setPosWorld (_x select 1);
    _vehicle setVectorDirAndUp [_x select 2, _x select 3];
    _vehicle enableSimulationGlobal (_x select 4);
    _vehicle setVariable ["ExileIsLocked", -1, true];

    if (_vehicle isKindOf "Exile_RussianRouletteChair") then
    {
        ExileRouletteChairs pushBack _vehicle;
        ExileRouletteChairPositions pushBack [_x select 1, getDir
_vehicle];
    };
}
forEach _vehicles;

// 0 Simple Objects
private _invisibleSelections = ["zasleh", "zasleh2",
"box_nato_grenades_sign_f", "box_nato_ammoord_sign_f",
"box_nato_support_sign_f"];
```

COPY   CLOSE

```
DMS_StaticMissionTypes =          [                         // List of STATIC missions with spawn chances.
                                      ["Sample_Mission",1],
                                      ["AI_Base",1],
                                      ["LHD_Raid",1],
                                      ["LHD_Occupation",1],
                                      ["Underground_Railroad",1]
                                  ];

DMS_BasesToImportOnServerStart =  [                // List of static bases to import on server startup (spawned post-init).
                                      "Sample_Objects"
                                  ];
```

Only if you have objects

```
DMS_BanditMissionsOnServerStart =  [

                                  ];

DMS_StaticMissionsOnServerStart =  [                         // List of STATIC missions with spawn chances.
                                      "Sample_Mission",
                                      "AI_Base",
                                      "LHD_Raid",
                                      "LHD_Occupation",
                                      "Underground_Railroad"
                                  ];
```

Obviously you will change the name of the Sample_Mission. Just be sure to ensure that all your commas are in the right place.

The static mission goes inside of the missions > static folder within DMS.

EXILE   EXILE

X. SUMMARY

This guide is geared for server owners that are using a rented server from a hosting company. You may apply some of this knowledge to a dedicated server, but not all. There will be additions in the future. If you need help with something that is not in my guide or just have further questions.

You can reach me on Exile or using my forums at www.teamkronik.com.

MORE TO COME