Buddy Bots Creation Tutorial

# Create Your Bot

Bots have 3 components: a def file, a main python file, and a state machine file

## Creating a Def File

Here is an example def file:

entityDef AssaultBot {

"inherit" "bot"

"author" "Dylan Rodarte"

"bot\_type" "Script"

"scriptclass" "AssaultBot"

"ui\_name" "AssaultBot"

}

To create a bot you need to change three of the variables.

1. Change the author to your name
2. Change the “scriptclass” and “ui\_name” to the name of your bot. These should be the same name
3. Change the name after entityDef

## Creating a Main Python File

The basic outline of the main file is as follows:

class DeathRowBot(afiBotBrain):

def Think(self , deltaTimeMS):

return botInput

def Spawn(self,spawnDict):

def Restart(self):

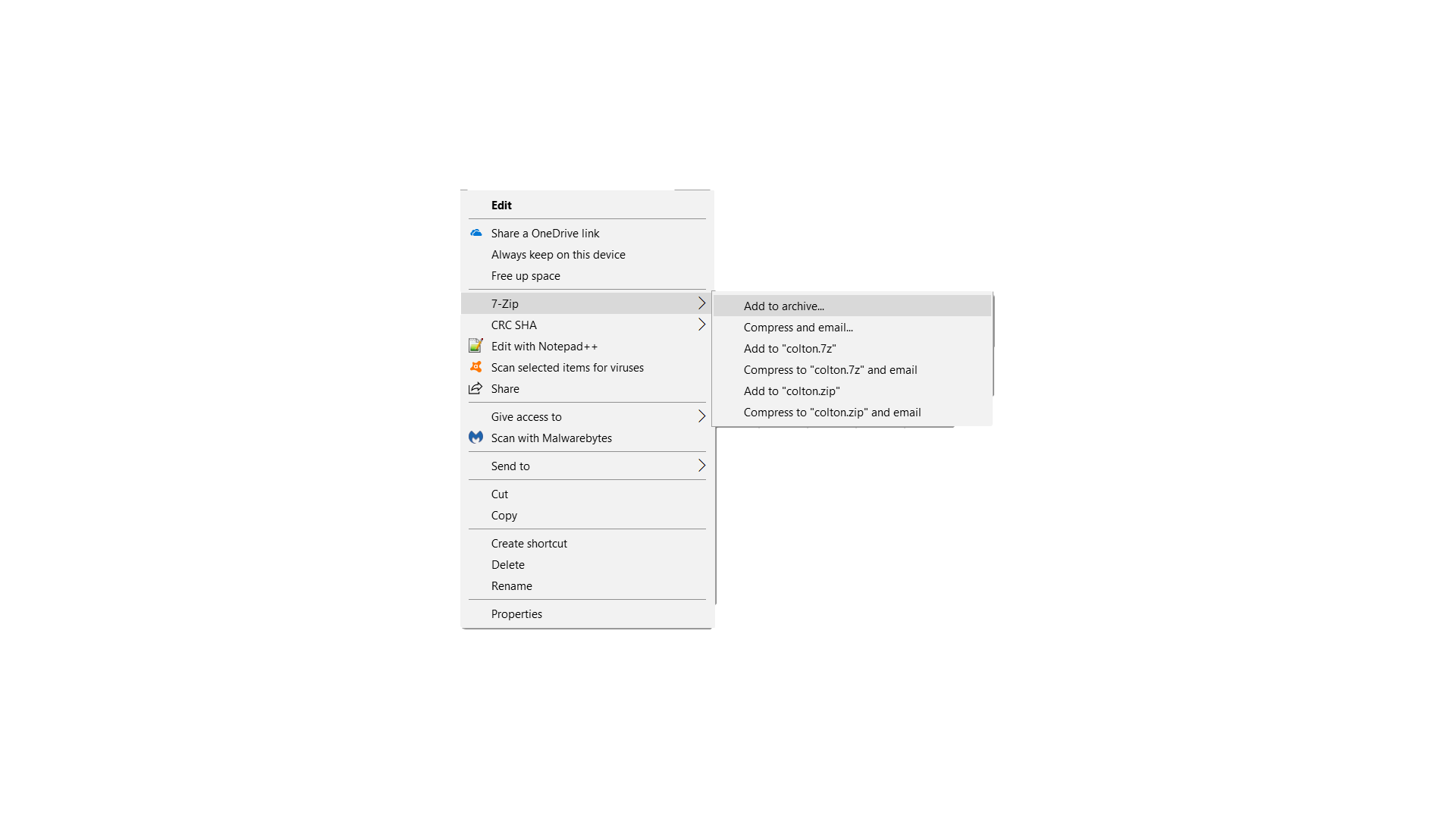
return

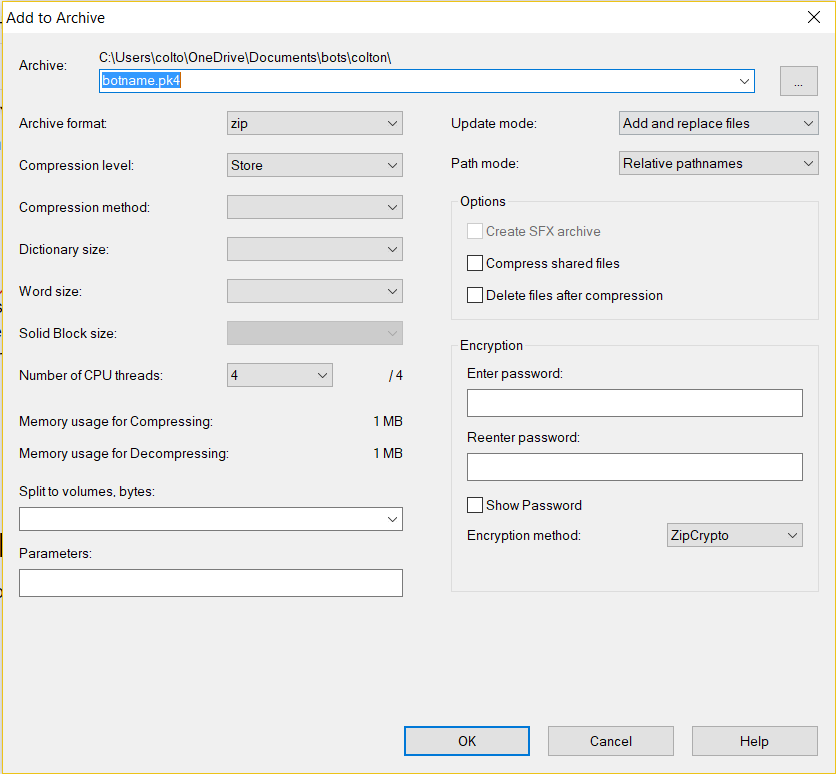
As you can see this bot is names “DeathRowBot” you should change the name to whatever name you put for “scriptclass” and “ui\_name”. Inside of Think and spawn youll put your code that controls the bot (see the api documentation for more available commands)

## Creating a state machine file

Simply copy an existing state machine file from one of the other bots

# 2. Pack up your bot

Youll need some kind of compression tool to place your files into a pk4. I recommend 7zip <https://www.7-zip.org/> its free and does everything youll need it to. Add the files to an archive and name the archive “nameofthebot”.pk4. When you compress the bot make sure to use the “zip” format and set the compression level to store



# 

# 

# 

# 

# 3. Place your bot in the botpaks folder

Copy your pk4 and place it into d3xp/botPaks

And youre done! To Spawn your bot enter a multiplayer game type “`” and type “addbot (TheNameOfYourBot). The following is going to be a short exercise to make a fairly simple example bot using the api and documentation. If you feel confident making a bot now you can skip it and just make your own.

Bot Creation Exercise

We’ll start with the basics, first, we need to import the necessary dependencies. Earlier we said to just copy and paste the “statemachine.py” from another bot. This is because they are all the same. The first three lines of all bots should be the same:  
  
import sys

sys.path.append("./d3xp/botPaks/Sample.pk4")

from statemachine import \*

This gets us all of the dependencies that are needed to make a bot “smart”.

The next step is to get the bot to shoot something. There are a lot of key functions that are needed to make a bot do anything. The first step is to use the “self” keyword.

We’re gonna need a list of bad guys to shoot. Additionally we’re gonna need to set up the initial input functions. Once again, we can just copy another two lines that will be in all bots.

class SampleBot(afiBotBrain):

def Think(self,deltaTimeMS):

botInput = aiInput\_t()

When we want to find nearby players we can simply call that function.

nearbyBotList = self.body.FindNearbyPlayers()

If there are nearby players then nearbyBotList will return true. If it’s true, we need to seek them and DESTROY them. You don’t, however, want to shoot your own team, so we need to get the team type of the nearby players. We can then get their health, their location and attack them.   
  
If the nearby players or bots are on our team, we need to move to find other bots. We use the same behavior to find bots if there are none around.

To do this we implement the following lines of code.

if nearbyBotList:

enemyBot = nearbyBotList.pop()

if self.body.team != enemyBot.GetTeam():

if enemyBot.GetHealth() > 0:

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.WALK )

self.body.MoveToPosition(enemyBot.GetPosition(),8)

self.body.LookAtPosition(enemyBot.GetEyePosition())

self.body.AmmoInClip()

self.body.Attack()

else:

self.body.StopAttack()

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.RUN )

else:

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.RUN )

self.body.StopAttack()

We can see above, that our bot is moving to the enemy bot, then looks/aims at the enemy bot, checks their own ammo supply, then shoots/attack. As seen above, the basic format is “self.body.DoSomething”.

It should be noted that there is currently no “roam” function. We can’t tell the bot to “walk forward 10 feet”. To get around this, we use the flag in the CTF mode. We can say “go get the flag but if you see anyone, shoot them”.

Now if we don't see any nearby players or bots, we have some freedom to choose what to do. We could just make a break for the flag or do something more fun.

In this simple example, we’ll “GET TO DA CHOPPA!” or flag in our case.

To do this, we need to check some things. We need to check if the enemy flag is in their base. If it is, then we run to go get it. If it is taken, then we should go defend OUR flag. The lines for this is pretty straight forward.

if afiBotManager.GetFlagStatus(self.enemyTeam) == flagStatus\_t.FLAGSTATUS\_INBASE:

self.enemyFlag = afiBotManager.GetFlag(self.enemyTeam)

self.body.MoveToPosition(self.enemyFlag.GetPosition(),8)

self.body.LookAtPosition(self.enemyFlag.GetPosition())

elif afiBotManager.GetFlagStatus(self.enemyTeam) == flagStatus\_t.FLAGSTATUS\_TAKEN:

self.ourFlag = afiBotManager.GetFlag(self.myTeam)

self.body.MoveToPosition(self.ourFlag.GetPosition(),8)

self.body.LookAtPosition(self.ourFlag.GetPosition())

return botInput

At the end, we return the bots decision as its input to the engine.

The last step is to handle the spawning of our bot. This isn’t that hard and can be done with the following lines of code.

def Spawn(self,spawnDict):

self.spawnDict = spawnDict

self.enemyTeam = 0

self.myTeam = self.body.team

afiBotManager.ConsolePrint("look mom, i made a bot")

str = "Confetti Team = {}\n".format(self.myTeam)

afiBotManager.ConsolePrint(str)

if self.myTeam == 0:

self.enemyTeam = 1

return

def Restart(self):

return

So in the end, bots aren't too complicated. Below is a completed bot from start to finish.

import sys

sys.path.append("./d3xp/botPaks/AssaultBot.pk4")

from statemachine import \*

class AssaultBot(afiBotBrain):

def Think(self,deltaTimeMS):

botInput = aiInput\_t()

nearbyBotList = self.body.FindNearbyPlayers()

if nearbyBotList:

enemyBot = nearbyBotList.pop()

if self.body.team != enemyBot.GetTeam():

if enemyBot.GetHealth() > 0:

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.WALK )

self.body.MoveToPosition(enemyBot.GetPosition(),8)

self.body.LookAtPosition(enemyBot.GetEyePosition())

self.body.AmmoInClip()

self.body.Attack()

else:

self.body.StopAttack()

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.RUN )

else:

self.body.UpdateAIMoveFlag( aiMoveFlag\_t.RUN )

self.body.StopAttack()

if afiBotManager.GetFlagStatus(self.enemyTeam) == flagStatus\_t.FLAGSTATUS\_INBASE:

self.enemyFlag = afiBotManager.GetFlag(self.enemyTeam)

self.body.MoveToPosition(self.enemyFlag.GetPosition(),8)

self.body.LookAtPosition(self.enemyFlag.GetPosition())

elif afiBotManager.GetFlagStatus(self.enemyTeam) == flagStatus\_t.FLAGSTATUS\_TAKEN:

self.ourFlag = afiBotManager.GetFlag(self.myTeam)

self.body.MoveToPosition(self.ourFlag.GetPosition(),8)

self.body.LookAtPosition(self.ourFlag.GetPosition())

return botInput

def Spawn(self,spawnDict):

self.spawnDict = spawnDict

self.enemyTeam = 0

self.myTeam = self.body.team

afiBotManager.ConsolePrint("There are many other flags, but this one is Mine!")

str = "Confetti Team = {}\n".format(self.myTeam)

afiBotManager.ConsolePrint(str)

if self.myTeam == 0:

self.enemyTeam = 1

return

def Restart(self):

return