

Forum:[Hack Forums](#)

Thread: [Netflix Account Checker](#)

Ninja - September 12, 2016, 3:39 pm

I was in need of some Netflix accounts because Narcos season 2 is out now, so I made an account cracker. Here Ill post the wrapper for Netflix only, not the rest of the program.

May seem like a ton of code for a few simple login functions, but it is for easy code maintainability and more OOP like structure. I have a pre-written model-controller format that I can copy and then edit. This also allows for easy integration into existing software. So don't hate and complain its over-complicated.

It re-uses sessions hence I didn't make the controller classes static. I recommend using 3 accounts per controller class and 1 proxy per controller class, else Netflix might catch on to you.

Requirements:

- .NET Framework 4.5

Usage: [\(Click to View\)](#)

Code using Netflix.Models;

using Netflix.Controllers;

```
void SomeMethod()
{
    WebProxy proxy = null;
    Account account = new Account("email", "password");
    AccountController controller = new AccountController(proxy);
    controller.Initialize();

    try
    {
        controller.PostLogin(account, proxy);
    }
    catch(LoginException lex)
    {
        Console.WriteLine($"Failed to login: {lex.Message}.")
    }
    catch(BadCredentialsException bcex)
    {
        Console.WriteLine("Invalid account credentials entered.");
    }
    catch(NoPlanException npex)
    {
        Console.WriteLine("Account has no active subscriptions.");
    }
}
```

[- GitHub -](#)

Zenco - September 12, 2016, 5:43 pm

thanks for this man, i'm sure it will help some people out :)

Ninja - September 14, 2016, 8:51 am

(Sep 12, 2016 05:43 PM)Zenco Wrote: [\[CITE\]](#) thanks for this man, i'm sure it will help some people out :)

Glad you liked it man! :)

redlightrouter - September 14, 2016, 2:59 pm

"Gekkevent". Sounds Dutch to me :D

Ninja - September 14, 2016, 3:27 pm

(Sep 14, 2016 02:59 PM)redlightrouter Wrote: [\[CITE\]](#) "Gekkevent". Sounds Dutch to me :D
Dat heb je goed gezien / You noticed well.

xYummy1337 - September 14, 2016, 10:20 pm

Thanks For This Sc's Man :333

Matrix Programmer - September 14, 2016, 10:25 pm

Holy shit. Using OOP for a Netflix Cracker. You have my respect. Not even joking.

#RESPECT

good job! tips fedora ?

SIM-1676753 - September 14, 2016, 10:26 pm

Tyrone check your Skype mate.

Sakuto - September 15, 2016, 2:16 am

(Sep 14, 2016 10:25 PM)Matrix Programmer Wrote: [\[CITE\]](#) Holy shit. Using OOP for a Netflix Cracker. You have my respect. Not even joking.

#RESPECT

good job! tips fedora ?

The only two programming paradigms that exist are OOP and Functional, everything else is shit.

Also, what were you expecting? Procedural in C#?

LastSession - September 15, 2016, 2:18 am

(Sep 15, 2016 02:16 AM)Sakuto Wrote: [\[CITE\]](#) The only two programming paradigms that exist are OOP and Functional, everything else is shit.

Also, what were you expecting? Procedural in C#?

xd just look at his threads

+Haze - September 15, 2016, 6:05 am

(Sep 15, 2016 02:16 AM)Sakuto Wrote: [\[CITE\]](#) The only two programming paradigms that exist are OOP and Functional, everything else is shit.

Also, what were you expecting? Procedural in C#?

Lol chill out it's a troll. Your too apprehensive and uptight around these coding sections. Take it easy.

Sakuto - September 15, 2016, 6:24 am

(Sep 15, 2016 06:05 AM)+Haze Wrote: [\[CITE\]](#) Lol chill out it's a troll. Your too apprehensive and uptight around these coding sections. Take it easy.

YOU ARE NOOB GTFO IM NOT SERIOUS.

Just kidding, I know, but you know, I keep the hope that one day, these sections will be HQ.

pechpilz - September 15, 2016, 1:23 pm

Thas nice.

Have a look at this.

Thanks 4 shareing :)

Ninja - September 15, 2016, 5:12 pm

(Sep 15, 2016 06:24 AM)Sakuto Wrote: [\[CITE\]](#) I keep the hope that one day, these sections will be HQ.

One can only dream...

Ninja - September 17, 2016, 11:38 am

Pushed out some updates.

The AccountController is no longer static and now re-uses sessions for faster speed.

nD_wuT - September 18, 2016, 10:34 pm

thank you for this! i might learn some new ideas

Ninja - September 19, 2016, 3:39 pm

(Sep 18, 2016 10:34 PM)nD_wuT Wrote: [\[CITE\]](#) thank you for this! i might learn some new ideas

Glad I was of help! :)

Benoit balls - September 20, 2016, 11:38 am

(Sep 19, 2016 03:39 PM)Ninja Wrote: [\[CITE\]](#) Glad I was of help! :)

Why not combine Account and AccountController? unless you were using with ASP.net?

Also why not catch WebException and check for response and maybe log it if it's an unexpected response?

I also didn't see a catch for WebException and checking if it was because of proxies.

also you should make GetSource an extension method for the HttpResponseMessage because later on you might want to use the response object.

Also last time I checked this isn't how you pass a parameter to a constructor : AccountController controller = new AccountController(WebProxy proxy);

Other than that, nice job, clean code.

MyRambo - September 21, 2016, 1:40 am

Thanks for the sharing this man ?

Ninja - September 21, 2016, 7:26 am

(Sep 20, 2016 11:38 AM)?Leonardo Da Vinci Wrote: [\[CITE\]](#) Why not combine Account and AccountController? unless you were using with ASP.net?

Why should I? I keep them separated as Account is just a simple class to hold data and AccountController can be used with multiple Accounts.

(Sep 20, 2016 11:38 AM)?Leonardo Da Vinci Wrote: [\[CITE\]](#) Also last time I checked this isn't how you pass a parameter to a constructor : AccountController controller = new AccountController(WebProxy proxy);

I had it there so people would know it is a Webproxy and not a string object that should be passed as a parameter. This was before I decided to add the line where I declare a proxy in the example.

(Sep 20, 2016 11:38 AM)?Leonardo Da Vinci Wrote: [\[CITE\]](#) Also why not catch WebException and check for response and maybe log it if it's an unexpected response?

If the user wants to he can do that himself.

There are no unexpected responses. WebExceptions only occur if you are using proxies that fail or if you are using no proxies at all.

(Sep 20, 2016 11:38 AM)?Leonardo Da Vinci Wrote: [\[CITE\]](#) also you should make GetSource an extension method for the HttpResponseMessage because later on you might want to use the response object..ReadResponse();

(Sep 20, 2016 11:38 AM)?Leonardo Da Vinci Wrote: [\[CITE\]](#) Other than that, nice job, clean code. Thanks.

+Haze - September 30, 2016, 4:20 pm

As we discussed on Skype these are my whole critiques.

1)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L52>

I see a scenario where an IndexOutOfRangeException is thrown

2)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L49>

Since you're parsing a web proxy, why don't you use a regex to validate the given string to see if it's a valid proxy. Afterall that is what Regex is for, that way you don't have to even check for any exceptions or if statements to check length of the array.

3)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L21>

Set Post what? The name of the method isn't even deceptive of what it does. SetPostBody() is more suitable.

4)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L25>

This also assumes that the content type will be "application/x-www-form-urlencoded", personally I think you should remove it and allow the user to set what they want instead.

5)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L26>

No need to do that, it automatically sets the count after you write to the stream

6)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L10>

Set Headers what? To me it seems more like you're initializing default headers. So in that case call the method InitializeDefaultHeaders()

7)

For all points previously made, you might say, "Yes, the developer can change it to suite there needs more specifically." But that shouldn't be the case, if you're following SOLID design principals then the open/closed principle states "software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification"; that is, such an entity can allow its behaviour to be extended without modifying its source code.

https://en.wikipedia.org/wiki/Open/closed_principle

8)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Models/Account.cs>

It's arguable but the Account data class is not really needed. Two parameters (string username, string password) would suffice. You're just writing extra code for nothing really. It seems to me like you're trying to overdo the OOP thing.

9)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L73>

I feel like you could of better designed this.

1 - Check what plan? Tonights plan? Tomorrows plan? Oh you mean Membership plan? Well you should make that clear in the method name.

2 - This should most definitely not be a void method, it should return a value. I feel like it should return a Enum type which has predefined values that represent the membership plans that netflix have.

So

Code `public MembershipPlan GetMembershipPlan() { }`
would of been a way better approach.

The reason I say this is because when the user/developer calls CheckPlan(), do they really wan to setup an ugly try catch block so I can tell their membership type? That's just horrible bad design.

10)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L78>

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L52>

DRY - Do not repeat yourself

11)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

Again, post login what?

The method name is fine, but it's not fine for a public method, it's better suited for a private method. One of the main principles of OOP is Abstraction, which is hiding the complexities of the inner workings of the system from the user by providing a simple interface that they interact with. So I feel as PostLogin should be a private method that is called by a public method the user can call. So

```
Code public boolean IsValidLogin(string username, string password)
{
```

```
    ....
    PostLogin();
    Do x with returned values....
    return boolean
}
```

12)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L23>

Why is this method public? It should be private and called from the class Constructor

13)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L36>

Why is the exception message empty? How is another developer supposed to know what went wrong

14)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L12>

Poor encapsulation over here, the set method should be private. The user outside shouldn't be able to alter it.

15)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

Like I said, the account class is very overkill. A simple, MethodName (string username, string password) would be fine.

16)

There is no sign of documentation at all. No commenting of methods, classes of any sort.

17)

Finally,

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

This should not be a void method neither. This should ideally return a value, weather it be a boolean, enum or custom class representing the status of the method call. Having it throw exceptions is pointless.

I'd rather be able to do

```
Code AuthenticationMessage authMsg = instance.AuthenticateUser("hi@gmai.com",
"heheHaze217");
```

```
If (authMsg.LoginSuccess)
{
    //Works
}
else
{
    Console.WriteLine(authMsg.ErrorMessage);
}
```

See how much neater that is? By doing this I don't have to create a new Expcetion class when I want to represent a new error message like you did.
than

```
Code try
{
    instance.PostLogin();
    //Login work
}
catch (BadCredentialsExcpetion e)
{
    //Invalid creds
}
catch (Exception e)
{
    //Not work
}
```

To wrap things up I think it's important to mention that sometimes its easy to forget these things as I tend to forget them sometimes, but when writing code you're releasing to the public, it's worth making the effort, since your employer could see the code when applying for jobs etc. Plus remember people learn from your code, so it's often good to show them something good to work with.

Ninja - September 30, 2016, 4:22 pm

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 1)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L52>

I see a scenario where an IndexOutOfBounds exception is thrown
Well im assuming the user is not a retard. Sorry for forgetting they usually are.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 2)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L49>

Since you're parsing a web proxy, why don't you use a regex to validate the given string to see if it's a valid proxy. Afterall that is what Regex is for, that way you don't have to even check for any exceptions or if statements to check length of the array.

I agree on the checking part but wouldn't regex be a bit over the top for this?

Just checking for 4 indexes should be good (of course not cover all errors and shit but still).

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 3)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L21>

Set Post what? The name of the method isn't even deceptive of what it does. SetPostBody() is more suitable.

Kill your own life.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 4)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L25>

This also assumes that the content type will be "application/x-www-form-urlencoded", personally I think you should remove it and allow the user to set what they want instead. Wat

So have it as a parameter or something? Its a simple class and it works for this. When I use it in other projects I manually change the string if needed...

Mons - September 30, 2016, 4:25 pm

This can be used for Netflix account generators?

Ninja - September 30, 2016, 4:25 pm

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 5)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L26>

No need to do that, it automatically sets the count after you write to the stream

Hmm didn't know that. Thanks.

+Haze - September 30, 2016, 4:25 pm

(Sep 30, 2016 04:25 PM)Mons Wrote: [\[CITE\]](#) This can be used for Netflix account generators?

no

but it can be used to check netflix accounts

Ninja - September 30, 2016, 4:25 pm

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 6)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Extensions/WebRequestExtension.cs#L10>

Set Headers what? To me it seems more like you're initializing default headers. So in that case call the method InitializeDefaultHeaders()

We gon bitch about method naming now boy?

(Sep 30, 2016 04:25 PM)Mons Wrote: [\[CITE\]](#) This can be used for Netflix account generators?

Yes and no. If you have an existing list of working accounts and want them double checked before

providing them to the end user then yes. But its a bit 'eh'.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 7)

For all points previously made, you might say, "Yes, the developer can change it to suite there needs more specifically." But that shouldn't be the case, if you're following SOLID design principals then the open/closed principle states "software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification"; that is, such an entity can allow its behaviour to be extended without modifying its source code.

https://en.wikipedia.org/wiki/Open/closed_principle

I was thinking about adding an optional parameter for a header collection and shit like that. But I am working on an alternative for webrequest based on sockets that should make everything easier and allow for socks support.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 8)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Models/Account.cs>

It's arguable but the Account data class is not really needed. Two parameters (string username, string password) would suffice. You're just writing extra code for nothing really. It seems to me like you're trying to overdo the OOP thing.

I C+P my initial design used for instagram where I would store other things in the account and provide them as a parameter to the controller functions. IE:

```
SetProfilePicture(Account account)
```

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 9)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L73>

I feel like you could of better designed this.

1 - Check what plan? Tonights plan? Tomorrows plan? Oh you mean Membership plan? Well you should make that clear in the method name.

2 - This should most definitely not be a void method, it should return a value. I feel like it should return a Enum type which has predefined values that represent the membership plans that netflix have.Or just a bool. I am just checking if it has a plan as in can view videos.

You're kind of right but I am too proud to admit so Ill just respond with the usual "kys faggot".

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 10)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L78>

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L52>

DRY - Do not repeat yourself

Didn't notice that. Thanks.

I mean... kys.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) DRY - Do not repeat yourself

DRY - Don't repeat yourself.

DNRY - Do not repeat yourself.

Check yo'self.

Fool.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 11)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

Again, post login what?

The method name is fine, but it's not fine for a public method, it's better suited for a private method. One of the main principles of OOP is Abstraction, which is hiding the complexities of the inner workings of the system from the user by providing a simple interface that they interact with. So I feel as PostLogin should be a private method that is called by a public method the user can call. So

```
Code public boolean IsValidLogin(string username, string password)
{
```

```
....
```

```
PostLogin();
```

```
Do x with returned values....
```

```
return boolean
```

```
}This also comes from the C+Ped design from my Instagram project where I would not check if a login was valid but just logged in and stored the session.
```

You're kind of right though on the naming. Adding a whole new method for it is kind of unnecessary don't you think?

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 13)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L36>

Why is the exception message empty? How is another developer supposed to know what went wrong

Hmm didn't see that.

I mean... kys.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 15)

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

Like I said, the account class is very overkill. A simple, MethodName (string username, string password) would be fine.

DRY - Don't repeat yourself.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 16)

There is no sign of documentation at all. No commenting of methods, classes of any sort.

Hence I provided a how to use in the OP.

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) 17)

Finally,

<https://github.com/GekkeVent/Netflix-Wrapper/blob/master/Netflix/Controllers/AccountController.cs#L46>

This should not be a void method neither. This should ideally return a value, whether it be a boolean, enum or custom class representing the status of the method call. Having it throw exceptions is pointless.

I'd rather be able to do

```
Code AuthenticationMessage authMsg = instance.AuthenticateUser("hi@gmail.com",  
"heheHaze217");
```

```
If (authMsg.LoginSuccess)  
{  
    //Works  
}  
else  
{  
    Console.WriteLine(authMsg.ErrorMessage);  
}
```

See how much neater that is? By doing this I don't have to create a new Excpetion class when I want to represent a new error message like you did.
than

```
Code try  
{  
    instance.PostLogin();  
    //Login work  
}  
catch (BadCredentialsExcpetion e)  
{  
    //Invalid creds  
}  
catch (Exception e)  
{  
    //Not work  
}  
}Didn't really think about using an enum. Thanks for the tip.
```

(Sep 30, 2016 04:20 PM)+Haze Wrote: [\[CITE\]](#) To wrap things up I think it's important to mention that sometimes its easy to forget these things as I tend to forget them sometimes, but when writing code you're releasing to the public, it's worth making the effort, since your employer could see the code when applying for jobs etc. Plus remember people learn from your code, so it's often good to show them something good to work with.

Im not giving my HF account to my employers good sir.

But yes, you are right in some points.

Now fuck off and let me live my life, mom!

I mean thank you...

+Haze - September 30, 2016, 4:43 pm

(Sep 30, 2016 04:22 PM)Ninja Wrote: [\[CITE\]](#) Well im assuming the user is not a retard. Sorry for forgetting they usually are.

I agree on the checking part but wouldn't regex be a bit over the top for this?
Just checking for 4 indexes should be good (of course not cover all errors and shit but still).

Kill your own life.

Wat

So have it as a parameter or something? Its a simple class and it works for this. When I use it in other projects I manually change the string if needed...

Not really, checking if there is 4 items in the array doesn't really suffice as hi:bye:hello:nice would pass when we know that isn't valid, which therefore makes your method unreliable. Regex is not overkill for this at all, this is what regex for designed for, pattern matching and data validation.

And yes manually changing the string means that the class wasn't well designed to suit your needs. Remember the Open Close princpal of SOLID design princpal. Yeah... Anyway other than that thanks for listening to my suggestions friend :)

Ninja - September 30, 2016, 4:50 pm

(Sep 30, 2016 04:43 PM)+Haze Wrote: [\[CITE\]](#) Not really, checking if there is 4 items in the array doesn't really suffice as hi:bye:hello:nice would pass when we know that isn't valid, which therefore makes your method unreliable.

I know, hence I said it doesn't cover everything that can go wrong. But if someone has a proxylist he usually checks them beforehand. And even if not, I have never found a list myself with letters in them.

(Sep 30, 2016 04:43 PM)+Haze Wrote: [\[CITE\]](#) And yes manually changing the string means that the class wasn't well designed to suit your needs. Remember the Open Close princpal of SOLID design princpal. Yeah... Anyway other than that thanks for listening to my suggestions friend :)

*BEST friend

and yes I know but Im not changing it as Im not going to be using the HttpWebRequest anymore in the future when Im done with my project.

Cairns - May 24, 2017, 7:52 am

I know this is kinda grave-digging but I stumbled across this thread and decided to go over your code and then read Haze's opinion on it. I just want to say that I agree with Haze on absolutely everything he has said, and you shouldn't take offense to it, afterall everyone lives a life to learn. But I also want to say you and others who have read his opinion should be grateful and learn from it. In all honesty his tips and his overall opinion on everything was great and accurate too. Not many developers here go into detail on what people should improve/fix and just bash people like little kids. Great job, Haze. Keep up the good work dude