

Forum:[Hack Forums](#)

Thread: [Very simple wrapper for the PasteBin API](#)

gigajew - April 22, 2017, 8:51 am

Just releasing this in case anyone wants to study or use it. It's a very easy-to-use wrapper for the PasteBin API. I created it for the user xCh3rx. It has the following methods:

```
PasteBin.Login(username, password);  
PasteBin.Paste(paste contents, public/private, subject, syntax type);
```

You can get your developer key after creating an account at <https://pastebin.com/api>

```
Code using System;  
using System.Collections.Specialized;  
using System.Net;  
using System.IO;
```

```
namespace HackForums {  
  
    public class Demo {  
  
        public static void Main(string[] argv) {  
            var username = ""; //username  
            var password = ""; //password  
            var developer = ""; //developer key  
  
            using(var pastebin = new PasteBin(developer)) {  
                if(pastebin.Login(username, password)) {  
                    var link = pastebin.Paste("Hello world"); //return pastebin link  
                    Console.WriteLine(link);  
                }  
            }  
        }  
    }  
}
```

//Created by GigaJew

```
public class PasteBin : IDisposable {  
  
    private WebClient Client { get; set; }  
  
    public string BaseUrl {  
        get {  
            return "http://pastebin.com/api/";  
        }  
    }  
  
    public string LoginUrl {  
        get {  
            return Path.Combine(this.BaseUrl, "api_login.php");  
        }  
    }  
}
```

```

    }
}

public string PasteUrl {
    get {
        return Path.Combine(this.BaseUrl, "api_post.php");
    }
}

public bool LoggedIn {
    get {
        return !string.IsNullOrEmpty(this.UserKey);
    }
}

public string DevKey { get; set; }

public string UserKey { get; set; }

public PasteBin(string devkey) {
    this.Client = new WebClient() {
        Proxy = null
    };
    this.DevKey = devkey;
}

public PasteBin() {
    this.Client = new WebClient() {
        Proxy = null
    };
}

public string Paste(string body, bool @private = false, string subject = "", string format = "text")
{
    if(!this.LoggedIn) {
        throw new Exception("Not logged in");
    }

    var response = this.Client.UploadValues(this.PasteUrl, new NameValueCollection() {
        { "api_dev_key", this.DevKey },
        { "api_option", "paste" },
        { "api_paste_code", body },
        { "api_paste_private", @private ? "0" : "1" },
        { "api_paste_name", subject },
        { "api_paste_expire_date", "N" },
        { "api_paste_format", format },
        { "api_user_key", this.UserKey }
    });

    using (MemoryStream memory = new MemoryStream(response)) {
        using (StreamReader reader = new StreamReader(memory)) {
            var link = reader.ReadLine();
            if(link.IndexOf("Bad") == -1) {

```

```

        return link;
    }else {
        throw new Exception(link);
    }
}
}

}

public bool Login(string username, string password) {
    if(string.IsNullOrEmpty(this.DevKey)) {
        throw new Exception("No developer key specified");
    }

    var response = this.Client.UploadValues(this.LoginUrl, new NameValueCollection() {
        { "api_dev_key", this.DevKey },
        { "api_user_name", username },
        { "api_user_password", password }
    });

    using (MemoryStream memory = new MemoryStream(response)) {
        using (StreamReader reader = new StreamReader(memory)) {
            var key = reader.ReadLine();
            if(key.IndexOf("Bad") == -1) {
                this.UserKey = key;
                return true;
            }else {
                return false;
            }
        }
    }
}

public void Dispose() {
    this.Client.Dispose();
}

}
}

```

Visual Freedom - April 22, 2017, 10:08 am

Sime yet effective. This is a good way to start with a pastebin integration.

Ninja - April 22, 2017, 11:20 am

Why a private property instead of a field? Just use a field. A property is used to 'protect' a field from outside sources.

Why are BaseURL, LoginURL, PasteURL. all public?

Why not set a bool for the LoggedIn property instead of a string nullcheck condition?