

Forum:[Hack Forums](#)

Thread: [C# Remote Method Invoking](#)

mynphit - April 21, 2012, 3:49 am

Hello HF!

This is my preview of my very basic and simple RemoteMethod class. It is currently able to invoke public static methods in public classes on the client (only works if the return value and the parameters are serializable), and return the return value to the server (via a encrypted tcp connection).

Thanks to [DragonHunter](#) for the Sync concept!

Video:

Spoiler ([Click to View](#))

[Video: <http://www.youtube.com/watch?v=OUOUiVbuZJk>]

Sorry for my English im Dutch

BaussHacker - April 21, 2012, 4:17 am

Wouldn't this be much easier?

<https://hackforums.net/showthread.php?tid=2338582>

Code public class WebLibrary

```
{
    private Dictionary<string, byte[]> libCollection;
    private string webUrl;

    public WebLibrary(Uri webUri)
    {
        webUrl = webUri.ToString();
        libCollection = new Dictionary<string, byte[]>();
    }
    public WebLibrary(string webUrl)
    {
        this.webUrl = webUrl;
        libCollection = new Dictionary<string, byte[]>();
    }

    public void Add(string LibraryName)
    {
        WebClient wc = new WebClient();
        byte[] data = wc.DownloadData(webUrl.EndsWith("/") ? webUrl + LibraryName + ".dll" : webUrl +
"/" + LibraryName + ".dll");
        libCollection.Add(LibraryName, data);
    }

    public void InvokeMethod(string LibraryName, string Class, string MethodName, object[]
```

Parameters = null)

```
{
    Assembly.Load(libCollection[LibraryName])
    .GetType(LibraryName + "." + Class).GetMethod(MethodName).Invoke(null, Parameters);
}
```

mysnphit - April 21, 2012, 4:57 am

(Apr 21, 2012 04:17 AM)?ray Wrote: [\[CITE\]](#) Wouldn't this be much easier?

<https://hackforums.net/showthread.php?tid=2338582>

Code public class WebLibrary

```
{
    private Dictionary<string, byte[]> libCollection;
    private string webUrl;

    public WebLibrary(Uri webUri)
    {
        webUrl = webUri.ToString();
        libCollection = new Dictionary<string, byte[]>();
    }
    public WebLibrary(string webUrl)
    {
        this.webUrl = webUrl;
        libCollection = new Dictionary<string, byte[]>();
    }

    public void Add(string LibraryName)
    {
        WebClient wc = new WebClient();
        byte[] data = wc.DownloadData(webUrl.EndsWith("/") ? webUrl + LibraryName + ".dll" : webUrl +
"/" + LibraryName + ".dll");
        libCollection.Add(LibraryName, data);
    }

    public void InvokeMethod(string LibraryName, string Class, string MethodName, object[]
Parameters = null)
    {
        Assembly.Load(libCollection[LibraryName])
        .GetType(LibraryName + "." + Class).GetMethod(MethodName).Invoke(null, Parameters);
    }
}
```

RemoteMethod is more dynamic, because the server can call all the static methods on the client side just by running the class, without downloading a dll or something, just with a encrypted tcp connection between the two pc's.

BaussHacker - April 21, 2012, 5:02 am

Oh wait, I misunderstooof OP, nevermind, just ignore my post.