

Instantly share code, notes, and snippets.



nasbench / Microsoft.NodejsTools.PressAnyKey.md

Created last year

☆ Star 0

🍴 Fork 0

<> Code

Revisions 1

Embed ▾

<script src="https://



Download ZIP

VisualStudio NodejsTools PressAnyKey Arbitrary Binary Execution

Microsoft.NodejsTools.PressAnyKey.md

Raw

Microsoft.NodejsTools.PressAnyKey.exe LOLBIN

This binary can be used as a LOLBIN as described [here](#).

Additional Info

- The arguments number must be at least 3
- The first first argument can be anything (instead of `both`, `normal` or `abnormal`). Since the `switch` clause doesn't specify a default case. And the `flag` variable is set to `true` before the check.
- The second argument also can be anything and it will be written to the execution path with the contents being the PID of the process `File.WriteAllText(args[1], process.Id.ToString());`
- The thrid argument is passed directly to `ProcessStartInfo` and is executed `Process.Start(startInfo);`. Hence anything can be called
- Any process launched from this, will be a child of `Microsoft.NodejsTools.PressAnyKey.exe`

Main Source

```
namespace Microsoft.NodejsTools.PressAnyKey
{
```

```
internal class Program
{
    private static int Main(string[] args)
    {
        if (args.Length < 3)
            Console.WriteLine("Usage: {0} (normal|abnormal|both) (pid file) (path to exe) [a]");
        Console.Title = args[2];
        ProcessStartInfo startInfo = new ProcessStartInfo(args[2], string.Join(" ", ((IEnumerable<string>)args).Skip(3)));
        {
            UseShellExecute = false
        };
        int num;
        try
        {
            Process process = Process.Start(startInfo);
            File.WriteAllText(args[1], process.Id.ToString());
            process.WaitForExit();
            num = process.ExitCode;
        }
        catch (Win32Exception ex)
        {
            Console.WriteLine("Failed to start process.");
            Console.WriteLine("Probable cause is the Node.js exe is corrupt, please re-install Node.js");
            Console.WriteLine("path: '" + args[2] + "'.");
            num = -1;
        }
        bool flag = true;
        switch (args[0])
        {
            case "both":
                flag = true;
                break;
            case "normal":
                flag = num == 0;
                break;
            case "abnormal":
                flag = num != 0;
                break;
        }
        if (flag)
        {
            Console.Write("Press any key to continue...");
            Console.ReadKey();
        }
        return num;
    }
}
```

```
}  
}
```

[Sign up for free](#) to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)



© 2024 GitHub, Inc.