Smuxi - Bug # 1072: exposed IpcChannel is world-writable

Status:	Closed	**Priority:**	Urgent
Author:	Mirco Bauer	**Category:**	Frontend GNOME
Created:	07/01/2015	**Assigned to:**	Mirco Bauer
Updated:	07/14/2015	**Due date:**	
Complexity:	Medium		
Found in Version:			
Subject:	exposed IpcChannel is world-writable		

Description:	in /tmp
	<pre>
	srw-rw-rw- 1 mirco.bauer mirco.bauer 0 2015-06-22 11:30 _usr_lib_smuxi_smuxi-frontend-gnome.exe
	</pre>
	
	This is problematic for systems with multiple users, as the other users can control the Smuxi instance of the first user.		

Associated revisions

07/14/2015 05:34 PM - Mirco Bauer

Frontend-GNOME: ensure rendezvous point for IPC is private (closes: #1072)

History

07/02/2015 12:09 AM - Mirco Bauer

From ./mcs/class/System.Runtime.Remoting/System.Runtime.Remoting.Channels.Ipc.Unix/IpcServerChannel.cs:

<pre>

```
internal static string BuildPathFromPortName (string portName)
{
    if (!Win32.IpcChannelHelper.IsValidPipeName (portName))
        throw new RemotingException ("Invalid IPC port name");
    return Path.Combine (Path.GetTempPath (), portName);
}
```

</pre>

07/02/2015 12:15 AM - Mirco Bauer

in mcs/class/Mono.Posix/Mono.Remoting.Channels.Unix/UnixServerChannel.cs:

<pre>

```
listener = new UnixListener (path);
Mono.Unix.Native.Syscall.chmod (path,
                Mono.Unix.Native.FilePermissions.S_IRUSR |
                Mono.Unix.Native.FilePermissions.S_IWUSR |
                Mono.Unix.Native.FilePermissions.S_IRGRP |
                Mono.Unix.Native.FilePermissions.S_IWGRP |
                Mono.Unix.Native.FilePermissions.S_IROTH |
                Mono.Unix.Native.FilePermissions.S_IWOTH);
```

</pre>

07/02/2015 12:19 AM - Mirco Bauer

from ./mcs/class/System.Runtime.Remoting/System.Runtime.Remoting.Channels.Ipc.Unix/README:

<pre>

System.Runtime.Remoting.Channels.Ipc.Unix
==========

*.Ipc.Unix is a wrapper for Mono.Remoting.Channels.Unix.
The Unix channels are loaded via reflection.

The wrapper is performing the following mappings:

IPC UNIX
---
portName="foo" path=$TEMP/foo
ipc://foo/bar.rem unix://$TEMP/foo?/bar.rem

</pre>

07/02/2015 12:35 AM - Mirco Bauer
So Mono is creating a unix socket in /tmp, chmodded 666, named after the port name of the IPC channel. According to MSDN docs [0] it is supposed to be only accessible by the same user by default, that would be chmod 600 on Unix.

[0]: https://msdn.microsoft.com/en-us/library/ms172351(v=vs.80)

07/05/2015 03:05 AM - Mirco Bauer
- Priority changed from Normal to Urgent

07/13/2015 08:20 PM - Mirco Bauer
Path.GetTempPath() -> [ICall] Path.get_temp_path() -> ves_icall_System_IO_get_temp_path() -> g_get_tmp_dir() of eglib [0] or glib [1]

[0]: https://github.com/mono/mono/blob/mono-3.2.8-branch/eglib/src/gmisc-win32.c#L146
[1]: https://developer.gnome.org/glib/stable/glib-Miscellaneous-Utility-Functions.html#g-get-tmp-dir

So the TMP, TMPDIR and TEMP environment variables influences the location of the chosen temp directory

07/14/2015 07:36 PM - Mirco Bauer
- Status changed from New to Closed
- % Done changed from 0 to 100
Applied in changeset commit:"709af0de6cdd439c307aeb359c7a309e2eede50e".