






# HttpGetFile

Function	HttpGetFile									
<b>Action</b>	<p>A built-in function for downloading content (resources) over HTTP(S) via the Tachyon Background channel, or from an external web-server.</p> <p>If the file cannot be downloaded, or the downloaded content does not match the hash and size supplied, an error occurs and the instruction is terminated.</p> <p>The file is downloaded into a randomly named sub-directory of TemporaryDirectory. From version 5.1 onwards, the location of <a href="#">TemporaryDirectory</a> is a configuration setting of the 1E Client.</p> <p>The sub-directory and file is deleted when the instruction ends. If you want to keep the file, make the instruction copy it to a permanent location.</p> <p>The following table shows the OS-specific defaults used if the setting is omitted, blank, or invalid. The default is recommended.</p> <table border="1"><thead><tr><th>Platform</th><th>Environment Variable(s)</th><th>Default</th></tr></thead><tbody><tr><td>Windows</td><td>TMP TEMP</td><td>%windir%\Temp which is the default %TEMP% location for SYSTEM.</td></tr><tr><td>Linux</td><td>TMPDIR</td><td>/tmp</td></tr></tbody></table>	Platform	Environment Variable(s)	Default	Windows	TMP TEMP	%windir%\Temp which is the default %TEMP% location for SYSTEM.	Linux	TMPDIR	/tmp
Platform	Environment Variable(s)	Default								
Windows	TMP TEMP	%windir%\Temp which is the default %TEMP% location for SYSTEM.								
Linux	TMPDIR	/tmp								
	<div style="border: 1px solid #ccc; padding: 5px;"><p> <b>Nomad integration</b></p><p>If 1E Nomad is available, it will optimize the download, storing a local copy of the file in the Nomad cache, with cache priority 1 and is subject to normal Nomad cache management.</p></div>									
<b>Parameters</b>	<p>URL (string): Either (a) the relative path to be appended to the "Content" directory under the Background Channel URL, or (b) a full URL starting with "http" or "https".</p> <p>It is a runtime error if the status code from a GET request on the full absolute URL is not 200.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"><p> "Content" is created as a virtual folder in IIS during installation of the Tachyon platform.</p></div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"><p> When an <b>absolute URL</b> is specified and the schema is '<b>https</b>' then the endpoint requesting the resource will attempt to verify the entire certificate chain of the resource as normal. This can fail if the endpoint does not recognize all the links in the chain of trust. The rules of checking are more rigorous than some browsers, which are often relaxed about the very root of the chain if it is signed using a <b>sha1</b> algorithm. This means whilst a browser may download the resource and display it, the endpoint might actually complain about the validity of the root Certificate Authority certificate.</p><p>To resolve this, you need to get hold of the public key for the root certificate and:</p><ul style="list-style-type: none"><li>• On Windows, it must be placed into the local machine trusted root certificate store</li><li>• On macOS, it needs to be in the 'System' folder of the Key Chain and explicitly trusted</li><li>• On other Unix/Linux-based endpoints that use a cacert.pem file, the root certificate should be exported as a base64-encoded CER format file and then the text appended to the cacert.pem file and the Agent restarted.</li></ul></div>									
	Hash (string): The SHA256 hash of the file. If the downloaded file's hash does not match this, a runtime error occurs.									
	Size (integer) : The size of the file in bytes. If the downloaded file's size does not match this, a runtime error occurs.									
	Suffix (string, optional) [introduced in v3.1] : Adds the specified suffix to the temporary file name. Usually this is not necessary, but some applications require a filename to end with a certain suffix. If the specified suffix does not start with a '.' then one is added automatically. Cannot be used with the PreserveName parameter.									
	Stagger (boolean, optional, default:true) [introduced in v3.2]: Instructs the download operation to either adhere to the configured random execution stagger setting if true, or cause it to download immediately (so no staggering) if set to false. If the agent is running on Windows AND Nomad is installed and available AND it is decided by other means that Nomad will be used, then staggering will not come into effect, regardless of the value, because Nomad will take care of not overloading the network.									
	PreserveName (boolean, optional, default:false) [Introduced in v3.4]: If false the directory will be the system temporary setting and the filename will be random. If true then directory will be some unique space for the execution of the containing instruction and the stem and suffix in the URL parameter are used as the filename. Cannot be used with the Suffix parameter.									

<p><b>Return values</b></p>	<p>FilePath (string): The full path of the downloaded file.</p> <div style="border: 1px solid red; padding: 5px; margin-top: 10px;">  This is a temporary file that persists only while the instruction is live. When the instruction ends the file will be deleted. If you want to keep the file, make the instruction copy it to a permanent location. </div>
<p><b>Example</b></p>	<p>A common use for this function is to download a script which can then be executed locally.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p><b>Example - download and execute a script</b></p> <pre> /* Download the script. Since we specify a partial URL (just the filename), the full URL is built by combining the specified URL with the root of the Tachyon Background channel. Note that because HttpGetFile is a built-in function, there is no need to specify a Module name */ @resource1 = HttpGetFile(URL:"MyCustomScript.ps1",                         Size:2137,                         Hash:" e0b5bb08b68ed4cdd73d035a775458e34bce1d68ec93a11defa62ca25e3d7037");  /* @script now contains a single row with one column named FilePath, so can be passed directly to the Scripting module */ @result = Scripting.Run(Language: "PowerShell", Script: @resource1, InterpretAsJson: true); </pre> </div> <p>To download content from an external web-server, you can instead set the URL property to the full URL of the file to be downloaded.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p><b>Example - downloading content from an external URL</b></p> <pre> /* Download file from web server */ @resource1 = HttpGetFile(URL:"http://webserver.acme.com/MyFile.jpg",                         Size:73355,                         Hash:" c9411ef157alb9e7d8418d0d9297b8df3f633b6alb2250d735c9b5cd32f1f795");  /* Build command line to copy temporary downloaded file to final location */ @cmd = SELECT "cmd.exe /c copy "    FilePath    " C:\\Pictures\\MyFile.jpg" AS CommandLine FROM @resource1;  /* Run the command built above */ NativeServices.RunCommand(CommandLine: @cmd.CommandLine);  /* Downloaded file will now be in C:\Pictures\MyFile.jpg */ </pre> </div>
<p><b>Platforms</b></p>	<ul style="list-style-type: none"> <li>• All</li> </ul>
<p><b>Notes</b></p>	<div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <b>Version tip</b>  Tachyon 3.1 uses a new URL format for retrieving downloaded files. Consequently, 3.1 agents are not fully backwards-compatible with 3.0 servers. They will communicate with them and execute simple instructions, but they will be unable to download content from a 3.0 server. </div>