



We use optional cookies to improve your experience on our websites, such as through social media connections, and to display personalized advertising based on your online activity. If you reject optional cookies, only cookies necessary to provide you the services will be used. You may change your selection by clicking “Manage Cookies” at the bottom of the page. [Privacy Statement](#) [Third-Party Cookies](#)

Accept

Reject

Manage cookies

Microsoft Ignite

Nov 19–22, 2024

Register now >



Learn

Discover >

Product documentation >

Development languages >

Topics >



Sign in

ⓘ We're no longer updating this content regularly. Check the [Microsoft Product Lifecycle](#) for information about how this product, service, technology, or API is supported.

[Return to main site](#)



Filter by title

Dynamic Host Configuration Protocol (DHCP)

About Dynamic Host Configuration Protocol

> DHCP Client API

> DHCP Server Callout API

DHCP Server Callout API

How the DHCP Server Callout API Operates

Chaining Multiple Third-Party DLLs

> DHCP Server Callout API Reference

> DHCP Server Management API

... / DHCP Server Callout API /



How the DHCP Server Callout API Operates

Article • 05/31/2018

Microsoft DHCP Server maintains an internal table of events that, upon their occurrence in DHCP protocol processing, trigger third-party DLL function calls. Developers register to have the occurrence of these events call their functions by creating a specified registry entry that Microsoft DHCP Server reads upon startup.

The location of the registry entry is:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\DHCPServer\Parameters

The following registry key values must be specified for this entry in order to enable the callout server:

Expand table

Key Name	Key Datatype	Value
CalloutDlls	REG_MULTI_SZ	String that contains the local path to the DHCP callout DLL. For example: "C:\Program Files\MyCalloutServer\my_dhcp.dll"
CalloutEnabled	DWORD	32-bit unsigned integer value that specifies 0 if the DHCP callout server is not enabled, and 1 if it is.

How Microsoft DHCP Server loads and handles third-party DLLs impacts third-party DLL developers. Microsoft DHCP Server takes the following steps in loading third-party DLLs:

1. Microsoft DHCP Server checks the previously defined registry location for the presence of third-party DLLs.
2. If no registry entries are found, the DHCP Server internal hook table remains empty, and no DHCP Server event notifications are sent.
3. If one or more registry entries is found, Microsoft DHCP Server reads the first registry entry, in alphabetical order, and attempts to load the corresponding third-party DLL. If the DLL loads successfully, Microsoft DHCP Server ceases checking for additional third-party DLLs.
4. Microsoft DHCP Server calls the [DhcpServerCalloutEntry](#) function in the loaded third-party DLL, retrieving the associated [DHCP_CALLOUT_TABLE](#) function and thereby

determining which events initiate notification to the third-party DLL. Notification comes in the form of corresponding third-party functions: one or more of which can be included in the third-party DLL, and each of which are defined in the following [DHCP Server API](#) reference pages.

Because the DHCP Server API enables developers to hook into the core functionality of Microsoft DHCP Server, use care when writing the third-party DLL functions. Several events that result in third-party DLL function calls occur when locks are held by the DHCP Server, and therefore, poorly written hooks can destroy DHCP Server heaps and critical information in Microsoft DHCP Server structures.