IP ADDRESS MANAGEMENT BEST PRACTICES FOR MICROSOFT NETWORKS

By: Brien M. Posey
Microsoft® DNS and DHCP services are widely used on networks large and small. This isn’t surprising considering that these services are included with Windows Server®, and are fully supported by Microsoft. Furthermore, Microsoft DNS and DHCP services have existed long enough to be stable and reliable.

Although Microsoft DHCP and DNS servers work well and do an undeniably good job, the free tools that Microsoft provides for managing these services lack scalability, alerting, reporting, multi-vendor support, and complete DDI functionality. As a result, Microsoft management utilities such as the DNS console, the DHCP console, and Microsoft IP Address Management (IPAM) tend to be adequate only for smaller organizations.

While Microsoft DNS and DHCP services work well, you may be looking for something with additional functionality. It may be advantageous to an organization to invest in a 3rd-party management tool, such as SolarWinds® IP Address Manager (IPAM) software. The 3rd-party tools exist with capabilities that far exceed those of native tools, and do so at a reasonable price. Such tools allow administrators to perform common management tasks more efficiently while allowing organizations to continue to leverage their existing investment in Microsoft services.

MICROSOFT OVERVIEW, LIMITATIONS, AND IMPACT

Microsoft has long acknowledged that keeping track of IP address usage is a challenge for network administrators. In Windows Server® 2012, Microsoft introduced its IPAM feature. Microsoft IPAM is a tool for centrally managing DNS and DHCP servers, and for managing IP address usage. Although Microsoft IPAM may prove to be adequate for some IT shops that use Microsoft DNS and DHCP servers exclusively, the tool has limitations that may prevent some organizations from using it.

Probably the most significant limitation is that Microsoft IPAM only works with Microsoft DHCP and DNS servers that are running on Windows Server® 2008 or higher. IPAM wasn’t introduced until Windows Server 2012, so the IPAM feature must run on Windows Server 2012 or Windows Server® 2012 R2. Microsoft IPAM is able to manage DHCP, DNS, domain controllers, and network policy servers running on Windows Server 2008 or higher (Windows Server 2008 R2 or higher is required for DHCP operational event auditing). Non-Microsoft servers cannot be managed using Microsoft IPAM.

Another limitation to using Microsoft IPAM is that the tool adheres to Active Directory® forest boundaries. The IPAM server itself must be domain-joined and can only monitor domain controllers, DHCP servers, DNS servers, and NPS servers that reside within the same Active Directory forest as itself. Microsoft IPAM is capable of monitoring servers residing in other domains regardless of whether or not those domains are trusted, but the monitored servers must be domain-joined.
Many of Microsoft IPAM limitations stem from its architecture. Microsoft supports Active Directory integration for its DNS servers. Similarly, DHCP servers are also authorized within the Active Directory. Hence, IPAM discovers DNS and DHCP servers and domain controllers by performing an Active Directory query. This query process is known as an IPAM server discovery. The Microsoft IPAM server discovery is manually initiated during the initial IPAM setup process and runs as a scheduled task after that. By default, the IPAM server discovery task is run once every 60 minutes, but can also be manually run on an as-needed basis. Domain controllers, DNS, and DHCP servers that are found by the Microsoft IPAM server discovery are automatically added to an IPAM server inventory list. Microsoft IPAM also supports the management of network policy servers (NPS), but NPS servers are not discovered automatically.

IPAM makes use of scheduled tasks, which means that the IPAM database is not kept up to date in real-time. This is why its architecture is considered a limiting factor. Updates made directly to DHCP servers are not reflected in the IPAM console until server data is retrieved either manually or by a scheduled task. Consequently, the IPAM console's view of the organization's IP address usage does not always reflect the current state. In fact, in some circumstances, it is even possible for two different IPAM servers to display conflicting IP address data. Microsoft IPAM servers do not share a common database, so if one IPAM server has run a data collection task more recently than the other, the data will be out of sync.

Microsoft IPAM is also lacking due to the fact that it does not include an alerting or reporting mechanism. This isn’t to say that the Microsoft IPAM console does not provide any useful data. It does. It’s just that Microsoft limits the ways in which that data can be used. Take the IP address ranges, for example. The Microsoft IPAM console allows you to view various configuration details, utilization trends, and other statistics on screen. You can filter this data and even create a saved query. Even so, there is no reporting mechanism per se. If you want to export IP usage data, your only options are to either use the console to create a CSV file, or write a custom Windows PowerShell® script to generate a report.

Not having an alerting mechanism causes problems because the Microsoft IPAM console displays conditions that need an administrator’s attention, but does nothing to call an administrator’s attention to those conditions. If you look at Figure A, for example, you see that four of the six IP address blocks are overlapping. Clearly, the console has done a good job of conveying this information. However, the console does not display any sort of message telling the administrator that there is a problem with their IP address blocks. It is up to the administrator to notice the information that is being displayed within the console.
The Microsoft IPAM console displays overlapping IP address blocks, but does not alert the administrator to the problem.

The previously discussed limitations can be thought of as inconveniences, rather than serious problems. The fact that Microsoft IPAM does not work across Active Directory® boundaries, for example, doesn’t mean that external forests can’t be monitored. It just means that administrators need separate IPAM deployments for each forest. It also means that there isn’t a good way of aggregating the data from the various IPAM servers. Still, these limitations should not be ignored. Operational inefficiencies stemming from Microsoft IPAM limitations can directly impact network support costs. For instance, if an organization is forced to deploy multiple IPAM servers in order to support multiple Active Directory forests, there is obviously a cost associated with deploying and maintaining those servers.

Similarly, Microsoft® IPAM has the potential to increase DHCP-related troubleshooting costs. On the surface, it would seem as though Microsoft IPAM should decrease DHCP-related support costs, as it can aggregate IP address data from across multiple DHCP servers. It is important to keep in mind, however, that the information displayed through the IPAM console is not always current. An administrator could conceivably be misled during the troubleshooting process simply because the data displayed within the IPAM console does not necessarily reflect the current state of the network.

In spite of its inability to display real-time data, the Microsoft IPAM console should theoretically make it easier for an experienced administrator familiar with the console’s nuances to troubleshoot a DHCP-related problem than it would be if they had to rely solely on the native DHCP console. Even so, there is a good chance that problems could be resolved even more quickly if the console were able to take automated steps to proactively prevent problems, and to alert the administrator when problems do occur.

Some of the operational issues surrounding Microsoft IPAM stem from the fact that Microsoft
IPAM is designed to monitor specific types of Microsoft servers (DNS, DHCP, NPS, and domain controllers). While system administrators are likely to see the value of using Microsoft IPAM to monitor Microsoft servers, Microsoft IPAM is likely to be inadequate for the needs of network engineers.

Network engineers need the ability to see the big picture with regard to network health. As such, network engineers might prefer an IP address management tool that integrates with the network management and monitoring tools they are already using. Even if a network engineer is willing to use Microsoft IPAM, its lack of support for non-Microsoft VM management tools, DNS and DHCP servers, and the way in which permissions must be granted for managing Microsoft DHCP and DNS servers, may be a deal breaker.

These issues become especially important when you consider that so many organizations are using VMware® as well as hardware routers to provide DHCP services to network endpoints. Even if an organization is currently using Microsoft Hyper-V® and DHCP servers, it is conceivable that they may wish to support other VM tools, or transition to hardware-based DHCP services in the future. Microsoft IPAM does not provide a migration path to such devices.

**WHAT ABOUT WINDOWS SERVER 10?**

Although the Windows Server® 2012 R2 version of Microsoft® IPAM is limited with regard to its capabilities, Microsoft has announced that its IPAM will be getting some new features and capabilities in Windows Server® 10. So far, none of the Windows Server 10 preview builds have included any of the forthcoming IPAM enhancements, but Microsoft has provided some information about what we can eventually expect (https://technet.microsoft.com/en-us/library/dn765487.aspx).

Perhaps the most useful of Microsoft’s planned improvements to IPAM is that the IPAM console will eventually support DNS management. Not only will IPAM support the discovery of DNS servers within an Active Directory® forest, administrators will be able to perform DNS management tasks within the IPAM console. Currently, administrators have to open Windows® DNS Manager as a second console if they want to create, edit, or delete DNS records.

Microsoft has also indicated that its IPAM will perform DNS resource record collections. This will allow Microsoft® IPAM to gather PTR records for reverse lookup zones. That way, if a reverse lookup zone is mapped to an IP address range, IPAM will be able to update the IP address inventory accordingly.

Another significant improvement planned for the next version of Microsoft IPAM is support for /31 and /32 subnets. These subnets are designated for certain purposes on some networks. For example, the /32 subnet allows for a single address, and can sometimes be used to provide loopback functionality for network switches. The /31 subnet allows for two addresses, and can be useful for site-to-site VPN.

In addition, Microsoft plans to introduce two new Windows PowerShell cmdlets. The Find-IpamFreeSubnet cmdlet can help an administrator determine which subnets are available for
use. Similarly, the new Find-IpamFreeRange cmdlet will be able to help an administrator locate free IP address ranges within a specified subnet.

While Microsoft will introduce some welcomed improvements to IPAM, the new feature set may not address all of Microsoft IPAM’s shortcomings.

**PROPOSED SOLUTION**

Microsoft IPAM server is better than spreadsheet-based IP address management. And, it may even be adequate for smaller organizations. However, the tool’s limitations make it impractical for managing larger networks.

Many 3rd-party IP address management products exist. But for a 3rd-party product to be a viable alternative to Microsoft IPAM, it must meet certain criteria, including:

- **The native ability to manage DHCP and DNS servers**
  A rip and replace solution that relies on proprietary DHCP and DNS servers may be impractical due to cost.

- **Support non-Microsoft products**
  While the network infrastructure may consist of mostly Microsoft server products, it may also include routers or other VM platforms and hardware appliances that perform DHCP services. Even if the organization is not currently using hardware-based DHCP services, using an IP address management solution with cross-platform support allows the organization to use solutions like VMware® and hardware appliances such as Cisco® DHCP or BIND DNS in the future if they want.

- **Reasonably priced**
  Microsoft IPAM is included for free in Windows Server®. While it is unrealistic to expect a 3rd-party solution to be free, it shouldn’t break the bank. Solutions that operate on appliances or require proprietary DNS and DHCP may be overly expensive.

- **Easy to use**
  Products with steep learning curves inhibit productivity, at least initially. A good litmus test is to ask how long it will take to get the product operational, and how many consultants are required.

- **Should (ideally) integrate with the organization’s other network management and monitoring tools**
  Although not technically a requirement, this is important because IP address management operates across many functional teams, including network engineering and administration.

**COMMON MANAGEMENT TASKS**

Administrators routinely perform a number of IP-related management tasks. Many tasks can be performed manually, using Microsoft® IPAM, or through a 3rd-party product such as SolarWinds® IP Address Manager. What follows is side-by-side comparisons of typical IP administration tasks using both Microsoft IPAM and SolarWinds IP Address Manager. We provide this comparison so
you can see the trade-offs required to use Microsoft IPAM vs. SolarWinds IP Address Manager to manage DHCP, DNS, and IP addresses.

**Task 1: Configuring IPAM**

We will start our discussion assuming you have successfully installed the applicable software, and you are now ready to configure it for use on your network. Our primary objectives are twofold: integrate applicable software with your DHCP and DNS services, and organize your IP space into subnets.

**Microsoft IPAM**

Microsoft IPAM does not use database sharing. Therefore, IPAM is unaware of which DNS, DHCP, NPS servers, or domain controllers are being managed by the IPAM servers on your network. You must enable the management of resources at the IPAM level. To do so, open Server Manager and select the IPAM option from the dashboard screen. There are six tasks you need to complete to prepare Microsoft IPAM for use.

**Task One:** Connect to an IPAM server. This is relatively easy to do. Click the Connect to IPAM Server link. Windows® displays a listing for your IPAM server. Select your IPAM server, and click OK.

**Task Two:** Provision the IPAM server. Click the Provision IPAM Server link. Windows® launches the Provision IPAM Wizard. This wizard is completely self-explanatory, and can be completed in a few clicks.

**Task Three:** Configure server discovery. Click the Configure Server Discovery link. When you do, Windows will display the Configure Server Discovery dialog box. You will need to select the domain within which you want to discover resources. You will also need to specify the types of resources you want to discover. By default, DNS servers, DHCP servers, and domain controllers are selected.

**Task Four:** Discover your servers. This is an automated task that you perform by clicking the Start Server Discovery link. The discovery process varies in time depending on the size of your network. For example, a large network may take a significant amount of time to complete.

**Task Five:** (to be completed only after server discovery is complete) Select or add the servers you want to manage. Click the Select or Add Servers to Manage and Verify IPAM Access link. Windows displays a list of the discovered servers.

Microsoft® IPAM allows DHCP servers to be brought under management by going to the Server Inventory tab and selecting either the Add Server or Retrieve All Server Data option from the Task menu. Doing so displays any resource that can be managed in the server inventory. Figure N shows a column named Manageability Status. By default, any detected servers are unmanaged and are blocked from being managed. It is up to you to unblock the server and bring it under
management. This is how you are able to control which IPAM server manages which server, and, therefore, how the IPAM topology will work as a whole. According to Microsoft, once Active Directory® synchronization completes, you should be able to refresh the display and the servers list as Unblocked. At that point, you should be able to set the Manageability Status to Managed.

Figure N

IPAM lets administrators choose which servers they want to bring under management.

After the discovered servers have been unblocked and the Manageability Status has been set to Managed, you can perform the sixth and final task.

**Task Six:** Retrieve data from the managed servers. Click the **Retrieve Data From Managed Servers** link. Upon completion of this task, you should be able to access IP address information through the Microsoft IPAM console.

To create one or more address blocks in Microsoft IPAM, begin by opening the IPAM console and navigating through the console tree to **IP Address Space | IP Address Blocks**. Next, select the **Add IP Address Block** option from the Tasks menu. This displays a dialog box similar to the one shown in Figure B, which asks for a network ID, prefix length, starting address, and ending address. You may also have to enter some other information if you choose to automatically assign address values. Click **OK** to create the block. The newly created IP address block does not appear in the console until you switch the current view to **IP Address Blocks**.
SolarWinds IP Address Manager

SolarWinds® IP Address Manager is able to manage IPv4 and IPv6 addresses, as well as DHCP and DNS services from Microsoft®, Cisco®, and open-source Internet Systems Consortium (ISC). SolarWinds IP Address Manager provides native two-way synchronization between its database and console and the managed service. For example, DNS entries made within SolarWinds IP Address Manager are pushed to the target DNS server, and changes made directly at the DNS server are synchronized and written back to the SolarWinds IP Address Manager Database.

When you first launch SolarWinds® IP Address Manager, you are automatically presented with a wizard to help you get started. Using this wizard, you can begin to define your IP space in one of three ways:

1. Automatically discover your subnets. This option scans router tables and automatically recreates your IP space within IP Address Manager.

2. Import an IP tracking spreadsheet. This lets you import your existing spreadsheets into IP Address Manager.

3. Create subnets manually. Use the wizard to name the subnet, and define it by IP range, or by using a CIDR mask.

The default choice is to discover your existing subnets. Use neighbor scans starting with the default gateway, and work out from there. Before the discovery scan can begin, SolarWinds IP Address Manager needs the SNMP credentials to the seed router(s) so it can access routing tables. In addition, SolarWinds IPAM needs to know how many router hops it can take in its discovery, and whether the default error-timeout values should be changed. Once the subnet discovery
scan is complete, you can review the discovered subnets and confirm which ones you want to import into SolarWinds IP Address Manager.

Once your subnets are defined, you can configure the subnet scanning feature. This provides an added layer of verification that reconciles which IP addresses are actually in use on the network with what has been recorded in the IP address manager database.

The SolarWinds IP Address Manager Getting Started Wizard also walks you through the process of adding your managed DHCP and DNS servers. If you use the Orion® network discovery service, adding a DHCP or DNS server is as simple as selecting from a list. Otherwise, if you do not want to use network discovery, you can provide the DNS name and/or IP address for each DHCP and DNS server to manage.

To use the network discovery service, go to the Summary screen and click Add DHCP Server or Add DNS Server and select Network Sonar Discovery. This starts a wizard that prompts you for a series of SNMP credentials, followed by a series of Windows credentials. The next screen asks you to enter the portions of your network on which you want to perform a network discovery. Although the wizard defaults to asking for an IP address range, you can instead select the Subnets tab and enter a series of subnets, as shown in Figure O. SolarWinds IPAM also gives you the option of specifying a seed router.

Click Next to access the options for configuring the discovery settings. The discovery settings consist of timeout values, retry counts, hop values, etc. Although these values are fully customizable, the default values are usually acceptable. Click Next. You will be prompted to either set a discovery schedule or perform the discovery immediately. After making your selection, click Discover and SolarWinds IPAM initiates the discovery process. The amount of time this process

Figure O

Enter the subnets you want to discover.
takes depends on the size of your network. When the discovery process is complete, click **Next** to acknowledge the devices that have been discovered. A list of resources to be managed (RAM, virtual memory, fixed disk, etc.) displays. Select the objects you want to manage, and click **Next**.

Review the list of objects to be imported, and click **Import**. The servers on the new subnet will be imported, as shown in Figure P.

**Figure P**

![Network Sonar Results Wizard](image)

The newly discovered servers have been imported into SolarWinds IPAM.

**Task 2: Deploy a new server**

One of the most common management tasks administrators perform is provisioning and deploying new servers (physical and virtual). The methods vary depending on whether the server will use a static or a dynamic IP address. In this example, we assume the server will receive a static IP address from an existing subnet.

**Microsoft® IPAM**

To assign a static IP to a server, navigate through the console tree to **IP Address Space | IP Address Inventory**. Select the **Add IP Address** command from the Tasks menu. Windows® displays the **Add IP Address** dialog box. At a minimum, this dialog box requires you to enter the IP address, the managing service (IPAM, DHCP, etc.), the service instance (Local Host), the device type (in this case, a host), the address state (in use), and the assignment type (static).

Because this is going to be a static IP address assignment, and we are using an IP address block that is not being used as a DHCP scope, we do not have to worry about creating a DHCP reservation. However, the console does give us the opportunity to create a DNS record for the new host. This DNS record consists of the device name, the forward lookup primary server, and the forward lookup zones. Reverse lookup zones are also supported, but are not required. Click
OK to add the new IP address. The IP address is added to the IP Address Inventory, as shown in Figure C. At this point, the new server can be provisioned with the new address and joined to the domain.

It is worth noting that this procedure reserves an IP address for the new server, and it creates a corresponding DNS record. However, Microsoft® IPAM does not actually check to see if the new server is using the address that has been reserved for it. Microsoft IPAM is limited with regard to comparing its IP address inventory to the network's actual IP address usage.

The new IP address has been added to the IP Address Inventory.

SolarWinds® IP Address Manager

SolarWinds IPAM takes a similar approach to finding an IP for a new server. From within SolarWinds IP Address Manager, click the Manage Subnets and IP Addresses menu. Use the navigation widget on the left to find the target subnet (or search for it). When the subnet is identified and selected, it lists an inventory of all IPs on the left side of the console. Set a view filter to display all Available addresses. To create an IP address reservation, select the desired address and click Edit. When the Edit IP Address dialog box appears, change the Status field from Available to Reserved. The software allows you to enter a host name, a comment, and other information for your reference. From here you can also create the DNS entries as part of the IP address provisioning process without having to complete a second step or use a second console. Click Save to create the reservation.
Figure D

The Manage Subnets & IP Addresses screen shows you which IP addresses are available and which are in use.

Task 3: Deploy a new printer

Another common management task is to provision and deploy endpoint devices typically managed by DHCP. In this example, we assume a printer is being deployed and will receive a DHCP-leased IP address.

Microsoft IPAM

There are a variety of methods for making a new printer available to clients on a Windows® network. The Print Management console included with Windows Server® can automatically detect all of the printers within its own subnet. Once a printer has been detected, Windows Server is able to automatically install the appropriate driver, set up the print queues, and share the printer with the end-users. The print drivers and printer connections can be made available to the users through group policy settings.

Before a new printer can be made available to Windows, it must be physically connected to the network and an IP address needs to be assigned to the printer. Many printers will attempt to contact a DHCP server and will automatically reserve an IP address. However, using a static IP address may be preferable.

As previously noted, Microsoft® IPAM is relatively accommodating with using static IP addresses. The first step in doing so is usually to reserve an IP address, either by creating a new IP address
range or by reserving an address from an existing DHCP scope.

Administrators wanting to reserve an address from an existing DHCP scope must determine which IP addresses are currently being used and which are available. For that, it is necessary to delve into Windows PowerShell®.

Suppose, for instance, you had an IP address range from 10.1.0.10 to 10.1.0.99. To check for free IP addresses within this IP address range, open Windows PowerShell and enter the following commands:

$Range = Get-IpamRange -StartIPAddress 10.1.0.10 -EndIPAddress 10.1.0.99
Find-IpamFreeAddress -InputObject $Range -TestReachability

You can see the output from these commands in Figure G.

Figure G

You can use Windows PowerShell to find an available IP address.

To reserve an IP address for a printer, make note of an available IP address that you wish to use. Next, navigate through the IPAM console tree to Monitor and Manage | DHCP Scopes. Now, right click on the scope that you want to modify, and select the Create DHCP Reservation command from the resulting shortcut menu. Windows displays the Create DHCP Reservation dialog box.

The Create DHCP Reservation dialog box asks for information about the address that you want to reserve. You will need to enter a name for the reservation (the name or a description of the printer is a good choice). Next, you need to enter the IP address that you had previously chosen, a client ID, and an optional description of the reservation. While you are at it, make sure that the Supported Type option is set to DHCP.

If you scroll through the Create DHCP Reservation dialog box, you will come to a section called DNS Dynamic Updates. In most cases, you want to disable dynamic DNS updates because printers do not tend to use Windows-style host names (although there are exceptions). When you finish
SolarWinds IP Address Manager
SolarWinds IP Address Manager's method of adding network printers is similar in its method of adding a new server. Again, from within SolarWinds IP Address Manager, click the Manage Subnets and IP Addresses menu. Use the navigation widget on the left to find the target subnet. When the subnet is identified and selected, it lists an inventory of all IPs on the left side of the console. Sort the Status column to find all available addresses. Select an available IP and click Edit. From here you can change the status of the IP address to Used and type to Static. Next, we will reserve this address within the DHCP scope defined for this subnet. From here, enter a host name, a comment, other information for your reference, and the DNS record entries, then click Save.

Task 4: Troubleshooting an IP address conflict
IP address conflicts can be especially troublesome for Windows® networks because an IP address conflict essentially causes a denial of service. In most cases, network devices with conflicting IP addresses cease to be able to participate on the network. Therefore, it is essential to be able to resolve IP address conflicts as quickly and efficiently as possible.

Microsoft IPAM
IP address conflict resolution can be extremely difficult if you are limited to using the native tools that are built into Windows Server®. Windows Server does not include tools that are dedicated to the task of resolving IP address conflicts. That being the case, conflicts must be resolved by using clues scattered across the IPAM console and various parts of the affected systems.

IP address conflict resolution is really made up of two separate tasks. The first involves physically locating the systems that are responsible for the conflict. This can be a tedious process. The second task is to determine the source of the conflict, which is usually relatively easy, but it’s often necessary to locate the affected systems first.

In some cases, you may be able to use tidbits of information found within the Microsoft® IPAM console as clues to the source of the IP address conflict. Keep in mind, however, that Microsoft IPAM is limited to working within an Active Directory® forest. As such, the Microsoft IPAM console isn’t much help if the conflict exists within devices that are not domain-joined. Furthermore, Microsoft IPAM does not associate an IP address and MAC address with the device’s make and model information, user information, switch/port location, and usage history.

To demonstrate the IP address conflict resolution process, we have brought online two new member servers running Windows Server® 2012 R2. The first of these servers has contacted a DHCP server that is being managed by Microsoft IPAM and has leased an address of 10.1.0.12. We then manually assigned the same IP address to the other member server. Windows® does not stop the conflicting IP address from being assigned.

The first symptom of a problem is that the member server with the manually assigned address
is unable to communicate on the network. At first, it would seem as though it would be obvious that an IP address conflict had occurred, and that you could easily fix the problem by assigning a new address to the server. However, keep in mind the fact that Windows does not display a pop-up error message indicating that there is an IP address conflict.

In this situation, an experienced administrator might immediately realize that an IP address conflict has occurred. By entering the IPCONFIG /ALL command into the Windows Command Prompt, you can confirm the existence of an IP address conflict, as shown in Figure H. This makes it easier to resolve the problem.

![Figure H](image)

*The IP address conflict can be confirmed through the Windows Command Prompt.*

In the real world, an administrator might not initially realize that an IP address conflict is the source of a connectivity problem. Unaware that an IP address conflict occurred, the administrator might attempt a variety of techniques to fix the problem.

Under normal circumstances, the server that acquired the address from the DHCP server should continue to function. It was the first server to use the address, and, therefore, Windows assumes that it is the rightful owner of the address. As such, only the server with the manually assigned IP address is prevented from communicating on the network.

With that in mind, imagine that the administrator decided to reboot the server with the manually assigned IP address in an effort to fix the problem. Let’s also pretend that around the same time the server that had acquired its address from the DHCP server were also rebooted, and that the reboots occurred in a way that resulted in the machine with the manually assigned address...
coming online first. In this case, the machine with the manually assigned address takes ownership of the address. The machine that has the dynamically assigned address eventually comes online, detects the IP address conflict, and acquires a different IP address from the DHCP server. In doing so, Windows has partially resolved the conflict.

From an administrator’s perspective, this automated conflict resolution might go completely unnoticed. After all, it is relatively common for problems to be cleared up after a reboot. Even so, a conflict still exists. Even though duplicate IP addresses are not currently being used on the network, there is a server that is using a manually assigned IP address that overlaps with a DHCP scope. This means that future reboots have the potential of causing another IP address conflict, resulting in the machine with the manually assigned IP address being unable to communicate on the network. Although the Windows Server® 2012 R2 networking stack takes steps to minimize the chances of this happening, servers are not the only potential source of an IP address conflict. An unauthorized address could just as easily be applied to a desktop, switch, router, or any other network device. Furthermore, Microsoft® IPAM does not check for IP address consistency with routers and switches.

This type of IP address conflict can be very difficult to troubleshoot for two reasons. First, Windows does not display an alert to make the administrator aware of the problem. Second, Microsoft IPAM does not display IP address information in real-time. Consequently, IPAM could theoretically provide the administrator with inaccurate IP address usage information unless the administrator knows to run a data collection task.

So what information can we get from the native Microsoft tools to assist with the IP address conflict? If we open the DHCP console and navigate through the console tree to DHCP | <DHCP server name> | IPv4 | Scope | Address Leases, we can see that 10.1.0.12 is listed as a bad address that is already in use, as shown in Figure I.

![Figure I](image)

The address is listed as already being in use.

The Microsoft IPAM console has an IP address tracking mechanism, but it isn't usually useful for troubleshooting IP address conflicts. You can access the IP address tracking feature by selecting the Event Catalog from the IPAM console tree, and selecting the By IP Address option from the lower pane. You can then enter the IP address in question along with the date range for the...
events that you are interested in. As you can see in Figure J, we searched on 10.1.0.12. Even so, there were no events listed that indicated an IP address conflict. All the events were related to either lease renewals or Kerberos tickets.

Figure J

![Image of IP Address Tracking](image1)

*The Event Catalog does not mention an IP address conflict.*

Another place where we can look for information is in the DNS console. The DNS console shows the host name for the last computer to associate a DNS record with the IP address, as shown in Figure K. On a small network this might be enough to help you to identify the machine that is using the unauthorized address. On larger networks, however, the host name is likely to be meaningless.

Figure K

![Image of DNS Manager](image2)

*The DNS console identifies the name of the host server that used the address most recently.*
Locating the machine that is using the IP address without authorization is likely to be a tedious manual process that involves checking individual machines one by one. The search can be narrowed down by network segment, and possibly by network name, but there isn't much more that can be done without the aid of 3rd-party tools.

**SolarWinds IP Address Manager**

SolarWinds® IP Address Manager takes a different approach to IP address conflict resolution. The product’s dashboard prominently displays IP address conflicts. As you can see in Figure L, SolarWinds displays the IP address for which the conflict has occurred, the address type, the subnet, the time when the conflict has occurred, the assigned MAC address, and the conflicting MAC address.

![SolarWinds IP Address Manager](image)

**Figure L**

SolarWinds IP Address Manager provides a great deal of information about IP address conflicts.

The information shown in the dashboard above provides a number of clues to help you locate the source of the IP address conflict. For instance, the Type field indicates the type of conflict that has occurred. A conflict might be attributed to a static address that conflicts with a dynamic address, or perhaps two DHCP scopes are overlapping. The time of conflict also provides clues, because the time stamp can be compared to the Windows® audit logs to determine what administrative tasks were being performed at the time of the conflict. This can help administrators determine which system might be the source of the conflict.

When you click on an IP address conflict, SolarWinds IP Address Manager displays more detail about the conflict. As you can see in Figure M, one of the devices is an IBM® System x3100 MS. We can see the system’s host name and the point of contact for the system. We can also see that
the other system involved in the conflict is a virtual machine. Not only can we see its host name, but we can view its IP address history. SolarWinds IP Address Manager even goes so far as to provide a recommended action. If optional SolarWinds® User Device Tracker (UDT) is installed, IP Address Manager takes UDT data and shows the user of the device, the switch port, and the switch the device is connected to. If you want to immediately remove the conflicting device from the network and eliminate the IP conflict, IP Address Manager lets you perform a remote port shutdown through UDT.

SolarWinds IP Address Manager provides detailed information about the IP address conflict.

The top left corner of the image above has three tabs. Right now, the IP Address Details tab is selected. But in the event that the administrator is unable to use these details to locate the machine that is causing the problem, they can use the User & Devices tab to determine who has been logging into the conflicting system recently.
Task 5: Identifying IPv6 enabled devices

As the number of available IPv4 addresses dwindles, organizations are likely to begin the transition to IPv6. In doing so, an organization must be able to determine which devices are IPv6-enabled.

**Microsoft IPAM**

Microsoft® IPAM supports both IPv4 and IPv6. In fact, a single IPAM server can manage up to 20,000 IPv4 addresses and an additional 20,000 IPv6 addresses. However, there are some limitations to Microsoft’s support for IPv6. Microsoft IPAM does not perform any special processing for IPv6 stateless address auto configuration private extensions. Similarly, Microsoft IPAM does not support auditing IPv6 stateless address auto configuration on an unmanaged machine to track the user. Likewise, Microsoft IPAM only provides IP utilization trends and only supports IP address reclamation for IPv4.

One of the limitations of the Microsoft IPAM console is that there isn’t a good way to view all of the current IPv6 addresses that are being used. The IPAM console's IP Address Inventory shows the static IP addresses that have been assigned through the IPAM console, but it will not show the IPv6 addresses that are actually being used across the network.

**SolarWinds IP Address Manager**

In contrast, SolarWinds® IP Address Manager makes it relatively easy to see the IPv6 addresses that are in use. From the Orion® Summary screen, click on the IP Addresses tab. When the Summary screen is displayed, scroll down until you see a section called Search for IP Address. Enter an asterisk into the Find field and click the Search button.

The results page shows all the IP addresses that SolarWinds IP Address Manager is aware of, but IPv6 addresses are not shown by default. To view IPv6 addresses, move your mouse over the column header. A down arrow icon should appear within the cell your mouse pointer is on. Click on this down arrow and you will see a menu, which contains an option to Add/Remove columns, as shown in Figure Q.

![Figure Q](image-url)

You will need to use the Add / Remove Columns option.
Select the Add/Remove Column menu’s IPv6 option. If you want to hide IPv4 addresses, you can deselect the Address option, as shown in Figure R. Because some search results may not contain IPv6 addresses, you may want to use the IPv6 column’s Sort Ascending or Sort Descending option to group all of the IPv6 addresses together.

![Figure R](image)

The results list now shows only IPv6 addresses.

**Task 6: Search for IP usage data**

It is important for an IP address management system to be able to provide detailed IP address usage information. Imagine that your organization detected a security breach and discovered that the attack came from a device located inside your firewall. It would be helpful to be able to determine which device was used in the attack, and which user was logged into the device.

**Microsoft IPAM**

Microsoft® IPAM is able to store up to three years of forensic data for up to 100,000 users within its own database. This data includes IP address lease data, host MAC address information, and user log on and log off data. It is worth noting that Windows® does not automatically purge aging data. It is up to the administrator to manually purge outdated data.

Forensic data can be examined through Microsoft IPAM’s event catalog. To do so, select the Event Catalog option from the console tree. The lower section of the console provides four different options for IP address tracking. Address usage can be tracked by IP address, by client ID, by host name, or by user name.

To track IP address usage by user name, click on the By User Name option and enter the user name and the target date range. Click the Search icon to display the search results.
Figure S, Microsoft IPAM does not always display data for the client ID, host name, or user name. This can make forensic analysis difficult, because it is challenging to use the IPAM console to determine which host a user was working from at the time of an incident.

Microsoft IPAM displays IP usage data, but it is typical for some data fields to be empty.

SolarWinds IP Address Manager
Like Microsoft IPAM, SolarWinds® IP Address Manager allows you to search for forensic information. To do so, go to the IP Addresses tab and click the **IPAM Summary** tab. Scroll down to the Search for IP Addresses section and enter your search query. You can search based on a variety of different criteria, such as alias, host name, IP address, comments, point of contact, DHCP client name, group description, group name, dual stack IPv6 address, and MAC address. You can choose to perform your search across multiple data fields, or limit your search to a specific type of data. As shown in Figure T, SolarWinds® IP Address Manager gives you the option of selecting which fields you want to include in your search.
SolarWinds IP Address Manager allows you to search across a number of different data fields.

At first glance, the results shown in Figure U might appear to be somewhat basic. However, a closer look reveals that there are ways of getting more information. For instance, you could select the result and click View Details or View Assignment History. The subnet name and the IP address listed within the result are also clickable, which allows you to drill further into the forensic data that is available.

SolarWinds IP Address Manager allows you to drill down into the search results in an effort to locate additional forensic data.
Although SolarWinds IP Address Manager does allow administrators to search for forensic information, this ability is greatly enhanced by the product’s ability to integrate with SolarWinds® User Device Tracker (UDT), which is able to provide detailed information on users, switch port usage, and wireless device usage.
CONCLUSION

Finding the right IPAM doesn’t need to be a huge investment in time or cost—or force you to halt your operations until you get what you need. With a little research, you can find a tool that aligns well with your system and infrastructure. That said, during your evaluation, you may find that many of the appliance-based, proprietary DHCP and DNS solutions include more functionality than you need, and cost more than you need to spend. On the other end, you may find the free tools, such as the free Microsoft® Server IPAM utility to be cumbersome, have a steep learning curve, or simply lack the functionality you need to adequately manage your IP addresses.

SolarWinds IP Address Manager complements a Microsoft-based infrastructure and offers the right incentives for considering a 3rd-party IP management tool. As observed in this paper, SolarWinds IP Address Manager completes most tasks faster and easier than the native server utilities. In addition, SolarWinds IP Address Manager offers these additional benefits:

1. Full DHCP, DNS, and IP management, which reduces work and improves accuracy.
2. Bi-directional overlay architecture, meaning that SolarWinds IP Address Manager works with Microsoft® as well as other vendors like VMware®, Cisco® and more.
3. DHCP, DNS, and IP alerting improves reliability by avoiding problems.
4. Works across multiple Active Directory® forests for greater visibility and control.
5. Integrates with SolarWinds network and systems monitoring.

Evaluating SolarWinds® IP Address Manager is easy and costs nothing. For more information, please visit http://www.solarwinds.com/lp/ip-control-bundle.aspx

ABOUT THE AUTHOR

Brien Posey is a freelance technology author with over two decades of IT experience, and has received Microsoft’s MVP award 13 times for his work with Windows Server, IIS™, Exchange™ Server, and File Systems / Storage. Posey has authored dozens of books and many thousands of articles, and routinely speaks at various international IT events. Prior to going freelance, Brien served as CIO for a national chain of hospitals and healthcare facilities. Previously, he served as a Network Engineer for the United States Department of Defense at Fort Knox. He has also worked as a Network Administrator for some of the nation’s largest insurance companies.