Troubleshooting DNS Problems in an Exchange Environment

Contributed by David Noel-Davies

DNS is often considered a “black box” in the sense that once you have it configured and working, it’s very hard to figure out why it breaks. Yet the health of your Exchange environment is tied to the health of your network’s DNS operations. There are three classes of DNS problems that can adversely affect Exchange: overall connectivity between Exchange servers and DNS servers, overall performance of DNS servers, and the DNS servers’ integrity—that is, the optimal configuration from both a technical and best-practices standpoint. Every DNS problem you encounter should fall into one of these categories. I’ll provide an in-depth look at DNS, its core components, and the approach you should take when troubleshooting specific classes of issues. Understanding the different classes of DNS issues and the troubleshooting process required to approach a specific type of issue will demystify DNS so that it’s no longer the black box it typically is for Exchange administrators. I’ll begin by looking at DNS’s related connectivity and performance issues, and in an upcoming article, I’ll explore DNS integrity.

Connectivity Issues

Troubleshooting connectivity entails testing whether the Exchange server can communicate with servers providing DNS services. Connectivity doesn’t focus on the overall application health of the DNS server—that is, whether or not DNS is responding to queries properly or whether zones exist and are correctly configured. Troubleshooting DNS connectivity is more about troubleshooting port connectivity and thus requires more knowledge about basic networking than the internals of the DNS application.One of the foundations of troubleshooting DNS connectivity is ensuring that both TCP and UDP ports 53 are available to your Exchange servers. A common mistake people make is assuming that only UDP port 53 is required. Although this is true for simple DNS queries (such as those made with Nslookup), Exchange requires both TCP and UDP for its DNS requests. When running basic connectivity tests to port 53, it’s important that you run these tests from the “perspective” of the Exchange server. In larger Exchange implementations, different firewall rules and zones might exist that could inhibit the ability of Exchange to contact its DNS servers, a situation that might go unnoticed when running tests from a workstation that traverses a different network path.Validating connectivity. Validating basic connectivity is only the first step in validating DNS connectivity. Once you’ve connected on TCP and UDP ports 53, you need to validate that the server is appropriately configured to receive DNS queries on these ports. In the case of Windows servers, unlike other platforms, DNS can’t be configured to listen on anything other than port 53. Note that if a rogue service is listening on port 53, the DNS service can’t start. So if the DNS service is started and you can connect via TCP and UDP to port 53, you’re well on the way to ensuring basic connectivity.The last thing that you’ll need to check for is making sure you have no IP conflicts on the network for the host servicing DNS requests. If there are conflicts, the DNS service might start successfully but might not be able to communicate with the network until the conflict is resolved. So, in order for your Exchange environment to pass DNS connectivity tests, the following four basic requirements must be met on your network:

- The DNS service should be started.
- The server should be able to receive traffic on TCP and UDP ports 53.
- There should be no IP conflict.
- There should be no port conflict. Note that the DNS service you’re using could be running either on Windows (the DNS Server service) or UNIX (the named—name server—daemon). If you can connect to your DNS service and are certain that the service you’re connecting to is really DNS, you can then begin to investigate how healthy the DNS service is.Troubleshooting Connectivity Issues

What should you do if you can’t connect? How you troubleshoot the problem depends on what connectivity tests fail. Can you connect on only TCP? Can you not connect at all? Have you validated the basics? Table 1: Possible Sources of DNS Connectivity Problems

Client

Server

Network

Port blocked

x

x

x

Routing

x

x

x

IP filters

x

RRAS filters

x

x

DNS not running

x

Port conflict

x

IP conflict

x

TCP/IP binding

x

x

DNS listening on IP

x Table 1 summarizes possible connectivity problems that are linked to DNS and where they originate (client, server, or network). It’s important to start your connectivity investigation locally before you investigate network problems. You can do so by performing a simple Telnet or Nslookup from the local server to itself. Note that by performing this test “locally” (without involving your network) you can eliminate the network from your initial troubleshooting. If for some reason these tests are failing and you’ve verified that your DNS Server service is started and has no IP or port conflicts, most likely a local software filter is causing the connectivity problems. Local software filters can include software firewalls, routers (e.g., RRAS), network card IP filters, and any other third-party software that could control traffic entering and leaving the network interfaces, including IPsec.System administrators are often quick to investigate and blame “the network” for Exchange problems, and often rightly so. Problems that arise from hardware configurations on devices such as firewalls, routers, or switches are commonly thought of as network issues. When you investigate hardware filter issues, it’s best to start by investigating the local network subnet, then expanding beyond that to remote subnets.Scenario 1: Exchange Servers on the Same Subnet
The simplest case is when the hosts are on the same subnet. When hosts are on the same subnet, devices such as firewalls and routers play no role in directing traffic between the devices. Only the switch or switches that the two servers are connected can determine whether traffic will be passed between the servers. If you know that the subnet mask is correct and the two servers still can’t ping each other, only a switch security configuration such as a Virtual LAN (VLAN) or ACL can prevent communication. Therefore, when you’ve narrowed down a DNS problem to servers on the same subnet, your troubleshooting should focus on security.Table 2: Networking Connectivity Problems Local Subnet Remote Subnet ROUTING Invalid subnet mask••Gateway downNA•Missing routeNA•Invalid gatewayNA•SECURITY Firewall ruleNA•Switch ACL••Router ACLNA•VLAN configuration•• Scenario 2: Exchange Servers on Remote Subnets
When servers aren’t on the same subnet, troubleshooting becomes more complex. In fact, the Invalid gateway item listed in Table 2 represents a number of possibilities. By using this table as a starting point, you can begin by first determining the scope under which you’re troubleshooting: local subnet or remote subnet. If your DNS server is on the same subnet as the DNS client (e.g., an Exchange server) that seems to be having DNS issues, you’d use the local subnet scope that Table 2 shows, for example.Once you’ve narrowed your scope, you can then focus your troubleshooting on the applicable components of your network. The gateway could be invalid simply because it’s inappropriate for the route or because its own gateway, routing tables, and other components are incorrect. You can perform a trace to the remote host, which can be very useful in determining where the problem lies. The trace

will tell you whether one Exchange server can connect to the destination network or a gateway that’s aware of the destination network. When troubleshooting a remote host, you should begin with testing that’s focused on the routing between your subnet and the remote subnet (we’ll assume the subnet mask is correct).Exchange server can connect to other hosts. If the DNS client (Exchange server) you’re troubleshooting can connect to other hosts on the remote network where the DNS server resides, the remote machine might have an invalid subnet mask or gateway settings. The remote host and/or ports might also have specific security settings that could prevent your Exchange server from connecting to it. If you can’t connect to any other hosts on the remote network, use a trace (if trace is enabled on your network) to validate how “close” you can get to the remote host. For example, if your network has five routers between two subnets, a trace might reveal that the connection drops at the third hop. This information would tell you that there’s likely a routing problem on the third-hop router. It might be missing a route to the next network or have an ACL glitch or other problems preventing traffic from getting to the destination network.Exchange server can connect to the network but not hosts. If you can reach the remote network but not the hosts, you’re likely dealing with a security-related problem. Probably specific ports are blocked. It’s also possible (but unlikely) that invalid subnet masks and gateways are set on all hosts. The gateway might also have an invalid subnet mask. If you fail to get to the remote host network, you should start “locally.” Look at your local routing table to see whether it includes any routes to the destination network. If the routing table has routes to the destination network, is the gateway you’re using reachable by ping or trace? Is it the correct gateway? Are subnet masks correct? If you don’t have routes to the destination network, you’ll be using your default gateway. Is it reachable? Is it the correct gateway? Are subnet masks correct?If your default gateway is reachable and correct but you can’t get to the remote network, you can perform a trace to find out which gateway is likely the problem. The gateway where the trace failed probably has a missing route, invalid gateway, or incorrect subnet mask.If you’ve validated your overall routing to the remote network and still can’t connect to the remote host, the problem might be security related. If possible, try connecting to the host on different ports or by using different methods. Ping is the most common and obvious method to test basic connectivity, but your network might block Ping (and trace). This might prompt you to use another method, such as Telnet, to test the connection. For example, if you know the machine is running Terminal Services, try performing a Telnet on port 3389. If you can successfully telnet to the host but you couldn’t ping it, you know that only DNS is blocked. Therefore, since the overall routing to the host network looks good and you still can’t connect, you can assume the problem is a security issue. Unfortunately, knowing exactly where the security-configuration problem lies will require you to investigate the specific configurations on your hardware device, which could be a firewall or router that you don’t have access to.Troubleshooting Performance Issues Optimizing performance is the goal that most administrators strive for but is hard to achieve. Your Exchange server’s performance is influenced by a variety of factors, ranging from hardware configurations to software parameters to application conflicts. To have a sense of what performance “should be,” it’s crucial that you understand your Exchange environment’s performance history. What is typical performance for the environment? Is typical performance in fact optimal? The first question is easy to answer, but determining the answer to the second often is an art.Baselines. Establishing a set of baselines is essential to knowing the typical performance patterns for a particular environment. It’s hard to ascertain whether performance is slow if you have no historical data to run comparisons against. In addition, without historical baselines, you’ll have difficulty determining whether the tuning you’ve done has successfully improved performance. Creating baselines for your Exchange environment involves more than just collecting basic information about major components such as processor or memory. You need to gather data that’s relevant to your DNS service.Measurement. It’s hard to know what to measure and when to measure items relevant to DNS performance. Measuring the well-known components of CPU, memory and disk I/O, is a good place to start. You can easily gather these metrics by using the Windows System Monitor (aka Performance Monitor) third-party monitoring products. However, if you’re measuring only these components, you won’t be able to detect when a problem originates with the DNS service or another component of the Exchange] server. Thus, it’s better if you can measure your CPU, memory, and disk I/O as they relate to the DNS service.Probably the most important (and relevant) quantity to measure on your DNS server in order to assess DNS performance in your Exchange environment is the DNS response time—that is, how quickly the DNS server responds to queries. Having this baseline statistic is important so that you can recognize when your DNS response time has dropped below normal. At a minimum, you’d want to know how your DNS service has performed over time and whether the CPU, memory, disk I/O, and response times are typical for your Exchange environment.Unfortunately there’s no performance counter called “DNS response time.” To measure DNS response time, you need to examine Windows System Monitor counters such as TCP Query Received/Sec and TCP Response Sent/sec and compare them. A high rate of queries received and a significantly slower response rate are clues that a performance problem might exist. You can use this same technique with UDP Query Received/sec and UDP Query Response/sec. When you start to see performance problems arise with either TCP or UDP queries (or both) you can further investigate the cause of these problems by examining the memory used by these requests and also possibly look into whether the number of such requests is unusually high or low.Again, knowing whether these numbers are significant depends heavily on your baselines. If the numbers are well above what you’ve historically seen, you know that you’re experiencing a performance degradation. A quick Google search will retrieve a list of several third-party tools that can help you measure and determine a baseline for your DNS response time. If you’re inclined toward scripting and want to get a sense of this without using such tools, you could write a script with an embedded DNS request that records a timestamp both before and after a DNS request finishes to assess response time for specific queries.Keep in mind that often other factors can adversely affect the DNS service’s performance. For example, a rogue process might consume CPU and cause the DNS server to seem “slow.” This isn’t a problem with the DNS service

directly but instead a problem with a rogue process that affects the DNS service. Equally important to consider is the interdependency of components and the chain reactions that performance problems on one component cause. For example, as the system runs out of memory, it starts to page the disk more, causing higher I/O. Understanding and considering these interactions on the server and its services is critical to troubleshooting DNS performance issues. You can find a full listing of counters that are important for DNS troubleshooting in the Microsoft article “Monitoring DNS server performance”.Troubleshooting. When you troubleshoot CPU-related issues, it’s essential that you examine requests received. An unusually large number of TCP or UDP requests can cause CPU problems. As I mentioned earlier this article, typically the UDP requests are performed by your average client, while TCP requests are used by Exchange. If you see a large number of TCP requests, this is a clue to investigate Exchange-related requests instead of troubleshooting simple lookups of host names, for example. In addition, zone-transfer requests can cause high CPU usage. An unusually large number of full-zone transfers can often cause performance problems when zones aren’t correctly configured. Some Exchange servers will request full-zone transfers instead of incremental transfers if the servers are out of sync. If you find that more zone-transfer requests are occurring than is typical for your Exchange organization, you’ll want to investigate your downstream DNS servers that are targets of these transfers as a potential cause of problems.You can identify memory issues for both TCP and UDP message memory use, which makes it easier to determine the type of client that’s causing memory-performance problems. Since we know that TCP DNS requests are made by Exchange servers, for example, issues that were solely TCP related would point us toward Exchange servers as the DNS clients to examine. Abnormal memory caching or database memory can be a clue to DNS-service memory problems. Disk I/O problems are usually directly related to CPU or memory problems that can cause the disk to write more information than usual. Therefore, often when you investigate a disk I/O issue, you should investigate CPU and memory issues in parallel.One common cause of DNS performance problems in larger environments is an unevenly distributed client load. In smaller environments, DNS performance problems might stem from other loads that are placed on a server running DNS along with multiple other applications. It’s crucial that your DNS infrastructure is designed to be scalable and that clients are configured to use DNS servers in a distributed fashion. This means that instead of having a “primary” and “secondary” DNS server in your organization, consider an architecture where one server services requests for half the clients in your organization, and the other server services requests for the other half. In this way, load is evenly distributed among the two DNS servers, preventing situations where the primary server is overloaded and the secondary server is used only when the primary is nonresponsive. Depending on the size of your environment, you might take different approaches to accomplishing a more balanced DNS server distribution for your clients. In fact, in smaller organizations this might not be a concern. However, if you decide that you want to more evenly distribute DNS load, you should make use of multiple DHCP scopes, which distribute different DNS data to client machines as well as ensure that servers with static DNS entries are set up in such a way that their load on DNS servers is also distributed.Getting a Grip on DNS Troubleshooting
DNS troubleshooting can be one of the more trying parts of an Exchange administrator’s job. But by following the approach that this article suggests and classifying DNS problems based on an issue’s particular symptoms, you’ll be better equipped to confront and conquer DNS problems in your Exchange environment.