SANS Institute
Information Security Reading Room

Preventing Windows 10 SMHNR DNS Leakage

Robert Upchurch

Preventing Windows 10 SMHNR DNS Leakage

GIAC (GSEC) Gold Certification

Author: Robert Upchurch, robert.upchurch@student.sans.edu
Advisor: Jonathan Risto

Accepted: February 25, 2021

Abstract

Microsoft enables Smart Multi-Homed Name Resolution (SMHNR) by default, sending name lookups out of all the connected interfaces for all configured name resolution protocols: DNS, LLMNR, and NetBIOS over TCP/IP (NetBT). Research on the effect that SMHNR has on DNS behavior showed that several users were concerned with DNS leakage ("DNS Leaks," 2017). DNS leakage is where unauthorized parties can observe, intercept, and possibly tamper with the name lookups or the lookup responses. Users were also frustrated by operational issues, such as attempting to resolve a private network hostname and receiving no response, a slow response, or an incorrect response while connected to a VPN ("Windows 10", 2015). This frustration led to users attempting to disable SMHNR ("Turn Off," 2021), but it did not always resolve the issue. The process to disable SMHNR varied based on the edition of Windows used, so the goal was to investigate the effect of SMHNR on DNS behavior and pursue an edition agnostic, native operating system method to mitigate that effect. Testing revealed that Name Resolution Policy Table (NRPT) rules provided a simple, scalable, and agile mechanism for controlling DNS client behavior that was effective across the multiple editions of Windows and worked irrespective of whether SMHNR was on or off.

1. Introduction

In 2012, Microsoft released the Windows 8 operating system. While ushering in a new look and feel that included cloud connectivity and a consistent user experience across a wide selection of devices, the Microsoft team simultaneously tried to improve performance, stability, and security. Indeed, it was no small task since performance and security tend to compete for priority. One of the features that the Windows 8 family of operating systems introduced was Smart Multi-Homed Name Resolution (SMHNR). In November 2013, Microsoft released some group policy settings that allowed for disabling the new SMHNR feature ("Windows 8", 2013). SMHNR's purpose was to improve system performance by accelerating DNS responses via name resolution optimization. Link-Local Multicast Name Resolution (LLMNR) and NetBIOS over TCP/IP (NetBT) are affected by SMHNR but are not relevant to this DNS leakage discussion.

Understanding the impact that name resolution optimization could have on security and privacy is crucial when deciding whether SMHNR is appropriate for the environment or situation. When mapped to the CIA triad of services: Confidentiality, Integrity, and Availability, SMHNR has the potential to disclose sensitive information, such as internal system names and geographic location of the system/user, to unauthorized parties via DNS leakage. The Integrity of the DNS results could be impacted, either intentionally or unintentionally, by receiving DNS answers from alternate or unexpected DNS servers. Unexpected or alternate answers could also disrupt the Availability of services, such as when needing to resolve names across a VPN tunnel. This research seeks to clarify how Smart Multi-Homed Name Resolution works in Windows 10, what native operating system settings to use to best control SMHNR DNS leakage, and the tests conducted to verify those settings. Finally, the goal is to provide enough information to evaluate the benefits and potential risks of SMHNR and offer a consistent, layered, edition agnostic approach for managing its effect on DNS behavior.

Robert Upchurch, robert.upchurch@student.sans.edu

2. Research

Testing solely included the IPv4 protocol and used a single DNS server configured on each interface. Deployed with Microsoft's latest image, all Windows machines used the default Microsoft configuration and were not domain-joined. Wireshark had the 'Time' column configured to show 'Time of day' and a display filter applied to show only relevant IPv4 DNS traffic.

2.1 Environment

The lab consisted of five virtual machines: 1 x 64-bit Windows 10-Home Edition, 1 x 64-bit Windows 10-Pro Edition, and 3 x 64-bit Kali Linux 2020.2. Each machine had three network interface cards configured for testing. Figure 1 illustrates the connectivity between devices.

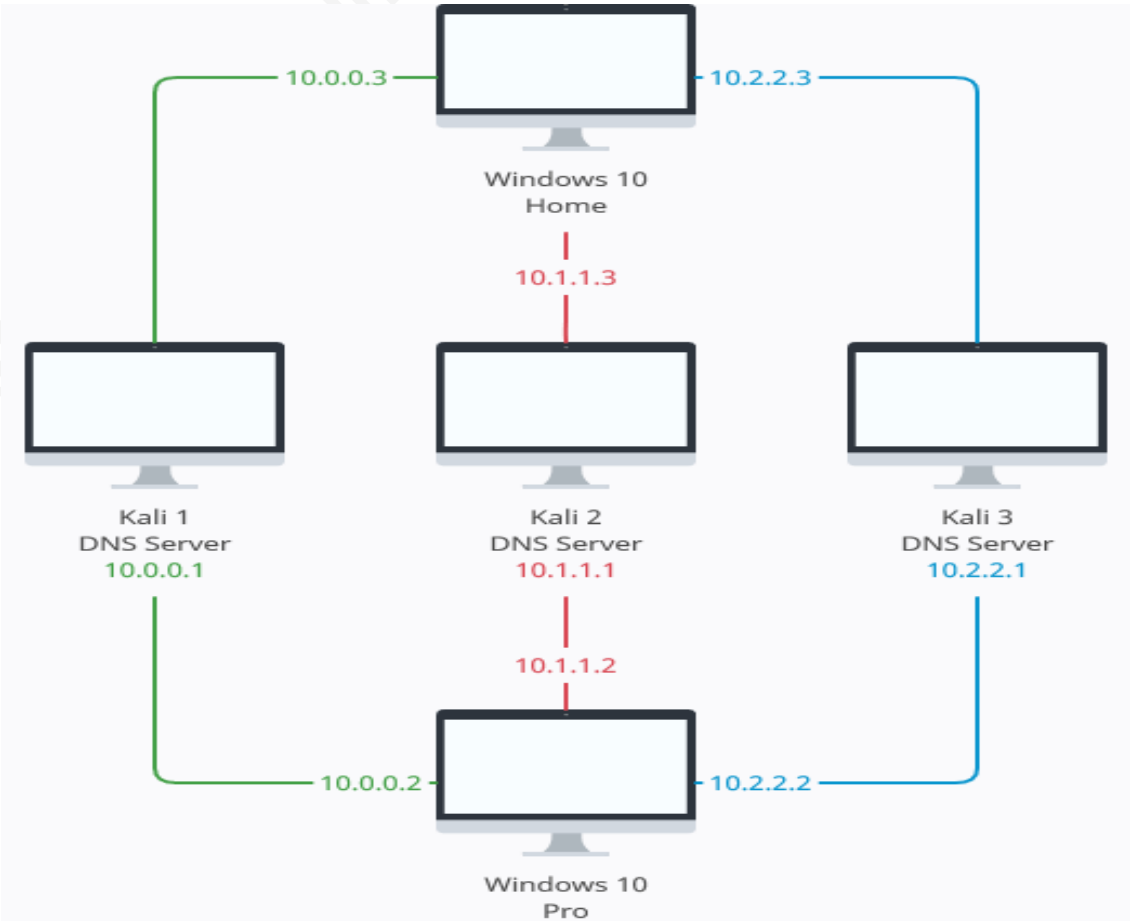

Figure 1. Lab Network Diagram

Robert Upchurch, robert.upchurch@student.sans.edu

Kali Linux machines were the DNS servers; each one was connected to a separate private network and configured with a different (A) record for www.example.com (Table 1).

DNS Server Name	DNS Server IP	(A) Record Host Name	(A) Record IP
Kali 1	10.0.0.1	www.example.com	10.0.0.1
Kali 2	10.1.1.1	www.example.com	10.1.1.1
Kali 3	10.2.2.1	www.example.com	10.2.2.2

Table 1. DNS Server to A Record Mapping

2.2 Methodology

Wireshark was used to monitor network traffic. Regedit and Powershell were used for configuration and testing since they were available in Microsoft's default image with no additional installation required. Each test included an evaluation with SMHNR on (Optimized) and SMHNR off (Non-Optimized). Tests ending with '.a' were conducted with SMHNR on (1.1.a, 1.2.a, etc.) and tests ending with '.b' were conducted with SMHNR off (1.1.b, 1.2.b, etc.). Each test began by clearing the DNS client cache.

The following outline describes the process used for

1. Interface Priority
   1.1. Control – Default Interface Metrics: *With all network adapters set to their default interface metric of 25, perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*
   1.2. Ethernet 2 – Lowest Interface Metric: *Change the Ethernet 2 interface metric to 20, keeping all other interfaces at their default settings. Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*
   1.3. Ethernet 3 – Lowest Interface Metric: *Change the Ethernet 3 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for*

hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.

1.4. Ethernet 4 – Lowest Interface Metric: *Change the Ethernet 4 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

2. DNS Response – Ethernet 4 Configured as the Preferred Adapter

2.1. Control – Positive DNS Server Response: *Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

2.2. NXDomain (No Such Name): *Perform a DNS query for hostname ftp.example.com, which does not have a name record; then, monitor the DNS server queried and the query order.*

2.3. (Ethernet 4) 2 Second Delay DNS Server Response: *Configure the preferred adapter's DNS server (Ethernet 4) to simulate a two-second delay. Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

2.4. (Ethernet 4) No DNS Server Response: *Disable the DNS service on the preferred adapter's DNS server (Ethernet 4) but keep the server online as it must be network reachable for this test. Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

3. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Configured as the Preferred Adapter

3.1. Control – No NRPT Rules Present (Default State): *With no NRPT rules present, perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

3.2. (Ethernet 3) Example.com Only Rule: *Create an NRPT rule that specifies the Ethernet 3 DNS server, 10.1.1.1, for the example.com domain name lookups without affecting any other lookups.*

3.2.1. *Perform a DNS query for hostname www.example.com; then monitor the DNS server queried, the query order, and the accepted DNS answer.*

3.2.2. *Perform a DNS query for hostname www.example2.com; then, monitor the DNS server queried and the query order.*

3.2.3. *Perform a DNS query for hostname ftp.example.com; then, monitor the DNS server queried and the query order.*

3.3. (Ethernet 3) Example.com and (Ethernet 2) Default Rule: *Leaving the example.com NRPT rule in place, create an additional NRPT rule that specifies the Ethernet 2 DNS server, 10.0.0.1, for all other lookups.*

3.3.1. *Perform a DNS query for hostname www.example.com; then, monitor the DNS server queried, the query order, and the accepted DNS answer.*

3.3.2. *Perform a DNS query for hostname www.example2.com; then monitor the DNS server queried and the query order.*

3.3.3. *Perform a DNS query for hostname ftp.example.com; then monitor the DNS server queried and the query order.*

3.4. (Ethernet 3) No DNS Server Response: *With both NRPT rules in place, disable the DNS service on DNS server 10.1.1.1, the example.com-specified lookup server, but keep the server online as it must be network reachable for this test.*

3.4.1. *Perform a DNS query for hostname www.example.com; then monitor the DNS server queried, the query order, and the accepted DNS answer.*

3.4.2. *Perform a DNS query for hostname www.example.com using the 10.2.2.1 DNS server by specifying the 10.2.2.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.3. *Remove all NRPT rules and perform a DNS query for hostname www.example.com. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.4. *With no NRPT rules in place, perform a DNS query for hostname www.example.com using the 10.0.0.1 DNS server by specifying the 10.0.0.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

Robert Upchurch, robert.upchurch@student.sans.edu

3. Findings

While researching Smart Multi-Homed Name Resolution, several websites posted about privacy concerns regarding the DNS leakage that SMHNR purportedly caused ("DNS Leaks," 2017). Tests performed as part of this research provided evidence detailing SMHNR's contribution to DNS leakage and what, if anything, could be done about it.

The following excerpt from Microsoft, "DNS Processes and Interactions" ("DNS Processes," 2013), describes the non-optimized (SMHNR Disabled) DNS behavior. Understanding this behavior is essential when testing whether optimized (SMHNR Enabled) DNS behavior creates an additional DNS leakage risk.

The DNS Client service queries the DNS servers in the following order:

1. *The DNS Client service sends the name query to the first DNS server on the preferred adapter's list of DNS servers and waits one second for a response.*
2. *If the DNS Client service does not receive a response from the first DNS server within one second, it sends the name query to the first DNS servers on all adapters that are still under consideration and waits two seconds for a response.*
3. *If the DNS Client service does not receive a response from any DNS server within two seconds, the DNS Client service sends the query to all DNS servers on all adapters that are still under consideration and waits another two seconds for a response.*
4. *If the DNS Client service still does not receive a response from any DNS server, it sends the name query to all DNS servers on all adapters that are still under consideration and waits four seconds for a response.*
5. *If the DNS Client service does not receive a response from any DNS server, the DNS client sends the query to all DNS servers on all adapters that are still under consideration and waits eight seconds for a response.*

If the DNS Client service receives a positive response, it stops querying for the name, adds the response to the cache, and returns the response to the client.

If the DNS Client service has not received a response from any server within eight seconds, the DNS Client service responds with a timeout. Also, if it has not received a response from any DNS server on a specified adapter, then for the next 30 seconds, the DNS Client service responds to all queries destined for servers on that adapter with a timeout and does not query those servers.

If at any point the DNS Client service receives a negative response from a server, it removes every server on that adapter from consideration during this search. For

example, if in step 2, the first server on Alternate Adapter A gave a negative response, the DNS Client service would not send the query to any other server on the list for Alternate Adapter A.

The DNS Client service keeps track of which servers answer name queries more quickly, and it moves servers up or down on the list based on how quickly they reply to name queries.

The group policy setting describes SMHNR's general behavior (Figure 2).

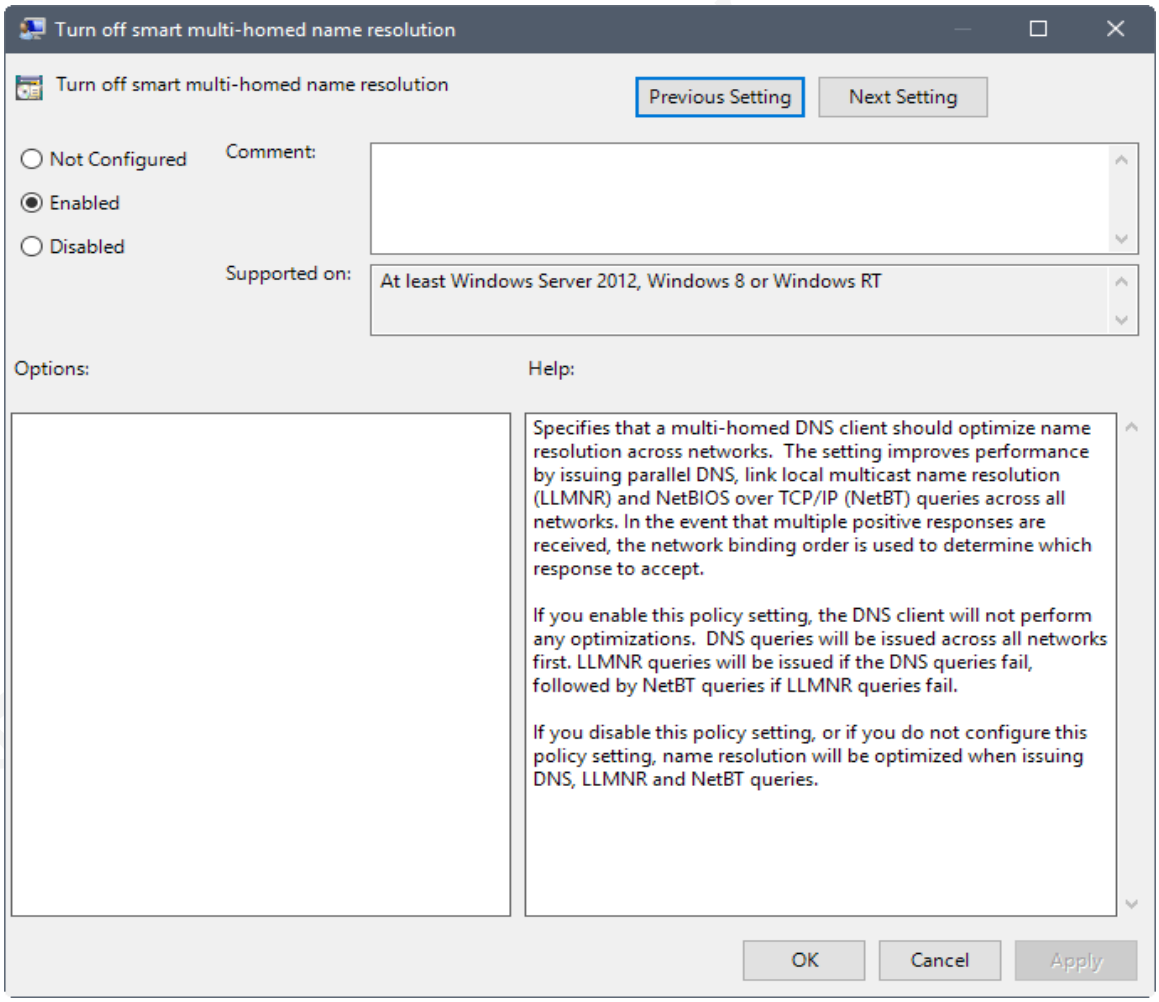

Figure 2. SMHNR Windows 10 Professional Edition - Group Policy Setting

Search results did not yield any Microsoft documentation that detailed the effect of SMHNR on DNS behavior. Observations during testing showed that it is similar to non-optimized behavior. However, optimized behavior starts by sending the DNS requests to all the first DNS servers for all SMHNR active interfaces (interfaces with default gateways configured). Optimized DNS behavior also seems to be more structured

in how it accepts responses. Non-optimized queries are sent out of all the interfaces if the first query fails, and once that happens, it accepts the answer that is received first. Both optimized and non-optimized send the queries simultaneously, from the highest priority (preferred) interfaces to the lower priority (alternate) interfaces. However, the SMHNR-optimized queries will wait for the preferred interface to timeout or receive a negative answer (NXDOMAIN). Then, they wait on the next highest priority interface, and they do not skip over a higher priority interface to accept a faster response.

In summary, there are two primary differences in behavior between non-optimized DNS queries and SMHNR-optimized DNS queries. First, non-optimized behavior will query the first DNS server on the preferred network adapter's interface. The 'preferred adapter' is the adapter with the lowest interface metric, making it the highest priority interface ("Configure The Order," 2020). If the first attempt does not work, it will query the first DNS server on all interfaces and, if there is still no answer, it will query all DNS servers on all interfaces. SMHNR-optimized DNS queries differ slightly from this behavior, starting at step two of the non-optimized process by querying the first DNS server on all interfaces first and progressing from there if no response is received. Second, if multiple positive responses are received, SMHNR will use the answer from the highest priority network adapter. In contrast, non-optimized uses the first response it receives regardless of interface priority. Sections 3.1 to 3.4 detail the tests used to reach these conclusions.

3.1 Interface Priority Tests

Purpose: The preferred adapter is supposed to determine the DNS server query order and, if multiple positive responses are received, the response that is accepted.

Control: Perform a DNS query for 'www.example.com' with all interfaces assigned their default metrics.

Hypothesis: The DNS server query order should change based on the interface's priority, from high to low, with the highest priority interface queried first and the highest priority positive answer accepted.

Tests:

1.1. Control – Default Interface Metrics: *With all network adapters set to their default interface metric of 25, perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

1.2. Ethernet 2 – Lowest Interface Metric: *Change the Ethernet 2 interface metric to 20, keeping all other interfaces at their default settings. Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

1.3. Ethernet 3 – Lowest Interface Metric: *Change the Ethernet 3 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

1.4. Ethernet 4 – Lowest Interface Metric: *Change the Ethernet 4 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

Tests Summary: Change an interface's metric to make it the preferred adapter, perform a DNS query for 'www.example.com', and monitor whether the DNS server query order changes, thereby changing the accepted DNS response.

Results:

Test #	Interface Metric (Ethernet 2,3,4)	Preferred Adapter (Ethernet 2,3,4)	DNS Query Order (Ethernet 2,3,4)	Accepted DNS Response (Ethernet 2,3,4)
1.1.a	25,25,25	Ethernet 2	Ethernet 2,3,4	Ethernet 2
1.2.a	20,25.25	Ethernet 2	Ethernet 2,3,4	Ethernet 2
1.3.a	25,20,25	Ethernet 3	Ethernet 3,2,4	Ethernet 3
1.4.a	25,25,20	Ethernet 4	Ethernet 4,2,3	Ethernet 4

Table 2. Interface Priority Tests (SMHNR ON)

Test #	Interface Metric (Ethernet 2,3,4)	Preferred Adapter (Ethernet 2/3/4)	DNS Query Order (Ethernet 2,3,4)	Accepted DNS Response (Ethernet 2/3/4)
1.1.b	25,25,25	Ethernet 2	Ethernet 2	Ethernet 2
1.2.b	20,25.25	Ethernet 2	Ethernet 2	Ethernet 2
1.3.b	25,20,25	Ethernet 3	Ethernet 3	Ethernet 3
1.4.b	25,25,20	Ethernet 4	Ethernet 4	Ethernet 4

Table 3. Interface Priority Tests (SMHNR OFF)

Results Summary: Each time the interface metric was changed, the interface with the lowest metric moved to the top of the binding order and became the preferred adapter, therefore becoming the first to send the initial DNS queries. Test 1.3 is an example of those metric and binding order changes. Changing the Ethernet 3 interface metric from its default of 25 to a metric of 20 (Figure 3) moved it to the top of the binding order list (Figure 4).

```
PS C:\Windows\system32> set-netipinterface -interfaceAlias "Ethernet 3" -int
erfaceMetric 20
PS C:\Windows\system32> get-netipinterface -addressFamily IPv4 |ft ifIndex,i
nterfaceAlias,interfaceMetric

ifIndex interfaceAlias             interfaceMetric
------- --------------             ---------------
      5 Ethernet 4                              25
      3 Ethernet 3                              20
      8 Ethernet 2                              25
      1 Loopback Pseudo-Interface 1             75
```

Figure 3. Set Ethernet 3 Metric and List Interface Metrics

```
PS C:\Windows\system32> route print
===========================================================================
Interface List
  3...08 00 27 6b 69 b9 ......Intel(R) PRO/1000 MT Desktop Adapter #3
  8...08 00 27 f4 bb ae ......Intel(R) PRO/1000 MT Desktop Adapter #2
  5...08 00 27 13 dc 8c ......Intel(R) PRO/1000 MT Desktop Adapter #4
  1...........................Software Loopback Interface 1
===========================================================================
```

Figure 4. Interface Binding Order After Metric Change

Additionally, suppose two or more adapters had the lowest metric, such as in the default state where Ethernet interfaces 2, 3, and 4 had the lowest metric of 25 (Figure 5). In that case, the preferred adapter could be identified by listing the binding order, which, in this case, was the Ethernet 2 interface (Figure 6).

```
PS C:\Windows\system32> get-netipinterface -addressFamily IPv4 |ft ifIndex,i
nterfaceAlias,interfaceMetric

ifIndex interfaceAlias             interfaceMetric
------- --------------             ---------------
      5 Ethernet 4                              25
      3 Ethernet 3                              25
      8 Ethernet 2                              25
      1 Loopback Pseudo-Interface 1             75
```

Figure 5. List Default Interface Metrics

Robert Upchurch, robert.upchurch@student.sans.edu

```
PS C:\Windows\system32> route print
===========================================================================
Interface List
  8...08 00 27 f4 bb ae ......Intel(R) PRO/1000 MT Desktop Adapter #2
  3...08 00 27 6b 69 b9 ......Intel(R) PRO/1000 MT Desktop Adapter #3
  5...08 00 27 13 dc 8c ......Intel(R) PRO/1000 MT Desktop Adapter #4
  1...........................Software Loopback Interface 1
===========================================================================
```

Figure 6. Interface Binding Order Before Metric Change

In this example, Ethernet 3 had the highest priority and was the first interface to send out the initial DNS query, regardless of whether SMHNR was on (1.3.a) or off (1.3.b). With SMHNR on, the initial DNS query went out all interfaces, but Ethernet 3 still sent the query first, as seen by looking at the Wireshark timestamps (Figure 7).

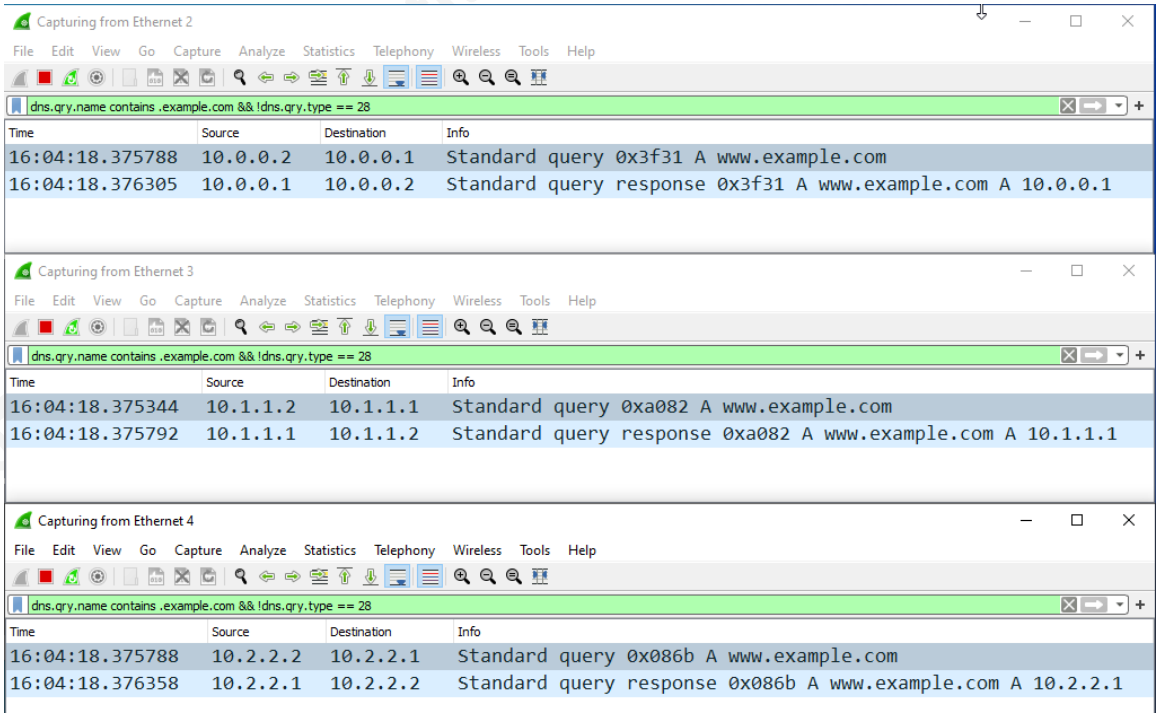

Figure 7. Wireshark Capture of DNS Query on all Test Interfaces (SMHNR ON – Ethernet 3 Preferred Adapter)

With SMHNR off, the initial query only goes out the Ethernet 3-preferred interface (Figure 8).

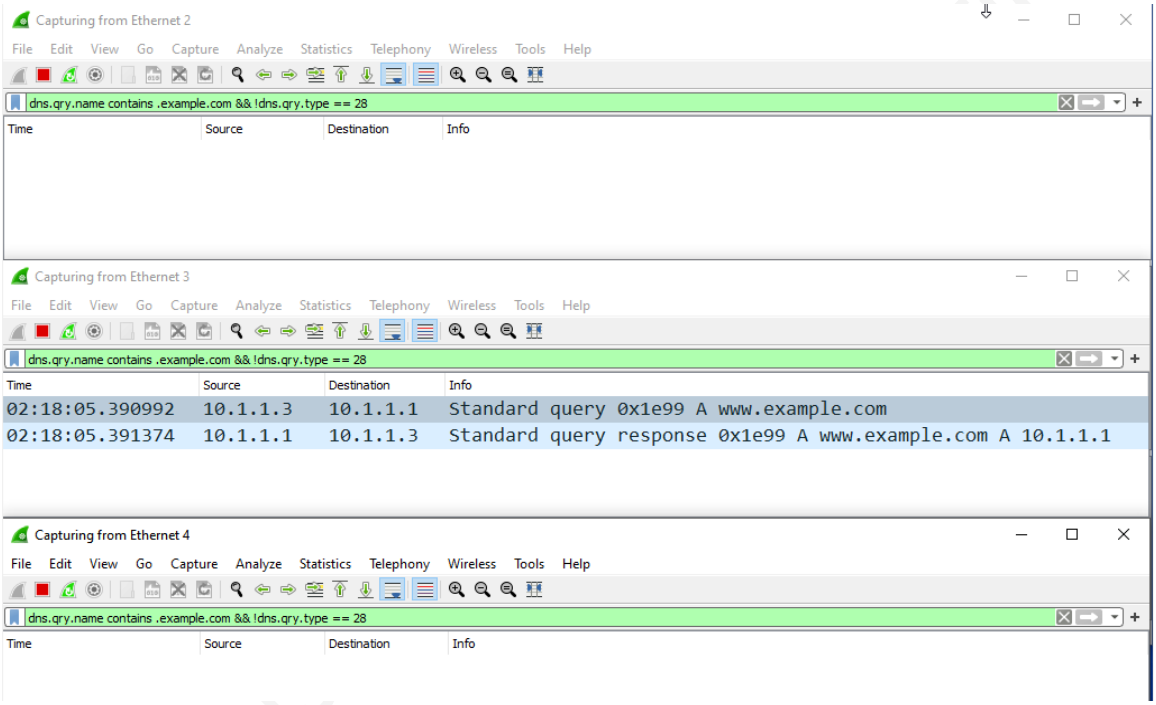

Figure 8. Wireshark Capture of DNS Query on all Test Interfaces (SMHNR OFF – Ethernet 3 Preferred Adapter)

3.2 DNS Response Tests – Ethernet 4 Preferred Adapter

Purpose: To discover what happens to the DNS query order and response acceptance if the DNS response is anything other than a successful, positive response (returns an IP address for the query).

Control: Perform a DNS query for 'www.example.com' and ensure that a positive response is received (IP address returned).

Hypothesis: The interface's binding order should determine the accepted DNS response. Suppose the preferred adapter (Ethernet 4) does not get a positive response, and the remaining interfaces do return positive responses. In that case, the DNS response received on the interface next in the binding order (after Ethernet 4) should be the accepted response.

Tests:

2.1. Control – Positive DNS Server Response: *Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

2.2. NXDomain (No Such Name): *Perform a DNS query for hostname ftp.example.com, which does not have a name record; then, monitor the queried DNS server and the query order.*

2.3. (Ethernet 4) 2 Second Delay DNS Server Response: *Configure the preferred adapter's DNS server (Ethernet 4) to simulate a two-second delay. Perform a DNS query for hostname www.example.com; then monitor the queried DNS server, the query order, and the accepted DNS answer.*

2.4. (Ethernet 4) No DNS Server Response: *Disable the DNS service on the preferred adapter's DNS server (Ethernet 4) but keep the server online as it must be network reachable for this test. Perform a DNS query for hostname www.example.com; then monitor the queried DNS server, the query order, and the accepted DNS answer.*

Tests Summary: Change the preferred adapter's DNS server records to replicate NXDomain (no such name), a two-second delay, and no server response types; then, query 'www.example.com' to see how the accepted DNS response behavior changes for each type.

Results:

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
2.1.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,3,2	Ethernet 4,2,3
2.2.a	ftp.example.com	Does Not Exist	Ethernet 4,2,3	Ethernet 4,2,3
2.3.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,2,3	Ethernet 3,2,4
2.4.a	www.example.com	10.0.0.1	Ethernet 4,2,3	Ethernet 3,2

Table 4. DNS Response Tests – Ethernet 4 Preferred Adapter (SMHNR ON)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
2.1.b	www.example.com	Eth 4: 10.2.2.1	Ethernet 4	Ethernet 4
2.2.b	ftp.example.com	Does Not Exist	Ethernet 4,2,3	Ethernet 4,2,3
2.3.b	www.example.com	Eth 3: 10.1.1.1	Ethernet 4,2,3	Ethernet 3,2,4
2.4.b	www.example.com	10.1.1.1	Ethernet 4,2,3	Ethernet 3,2

Table 5. DNS Response Tests – Ethernet 4 Preferred Adapter (SMHNR OFF)

Results Summary: The hypothesis was correct when SMHNR was on; however, when SMHNR was off, the first response received was accepted; regardless of the interface binding order. Test 2.3 illustrated that, with SMHNR on (2.3.a), if the preferred adapter received a positive DNS response after the other interfaces (Figure 9), it still used the answer 10.2.2.1 from the preferred adapter as the accepted response (Figure 10).

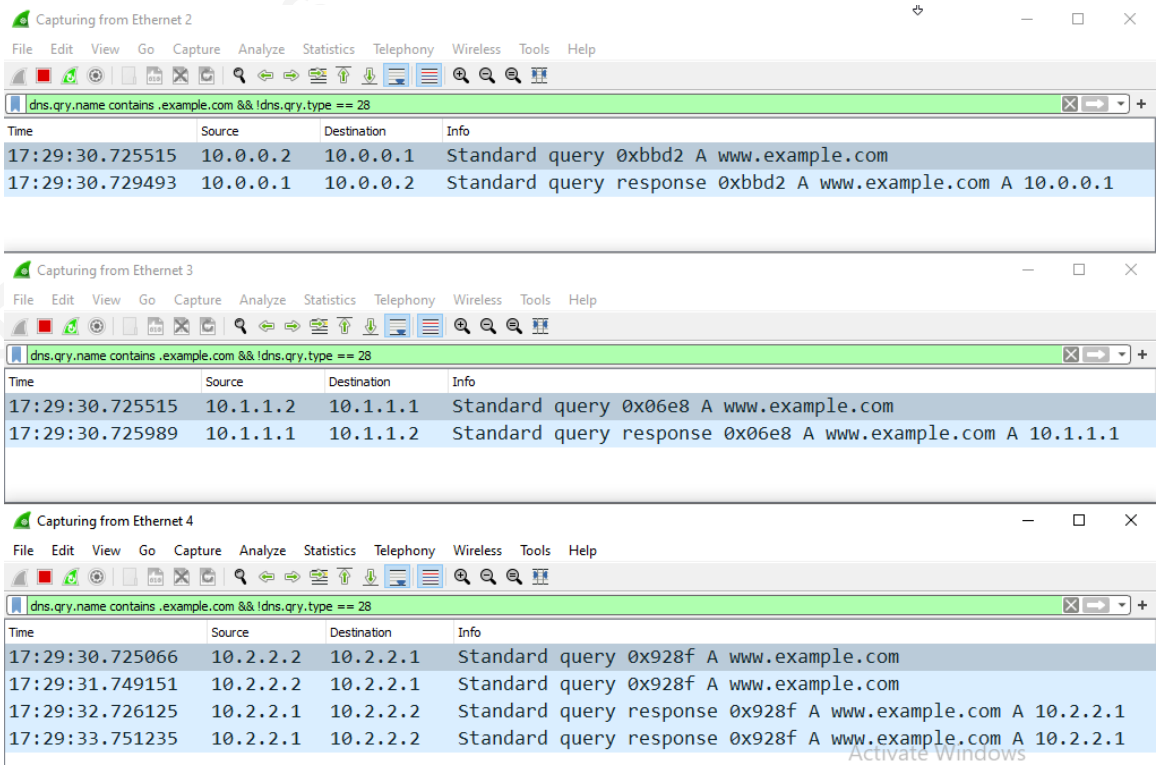

Figure 9. Wireshark Capture – Ethernet 4 Delayed Response

```
PS C:\Windows\system32> Clear-DNSClientCache
PS C:\Windows\system32> Resolve-DNSName www.example.com |FL

Name       : www.example.com
Type       : A
TTL        : 86400
DataLength : 4
Section    : Answer
IPAddress  : 10.2.2.1
```

Figure 10. Ethernet 4 Accepted Response

When SMHNR was off (2.3.b), the first positive response received, 10.1.1.1 (Figure 11), became the accepted response (Figure 12), despite Ethernet 2 being higher in the interface binding order (Figure 13).

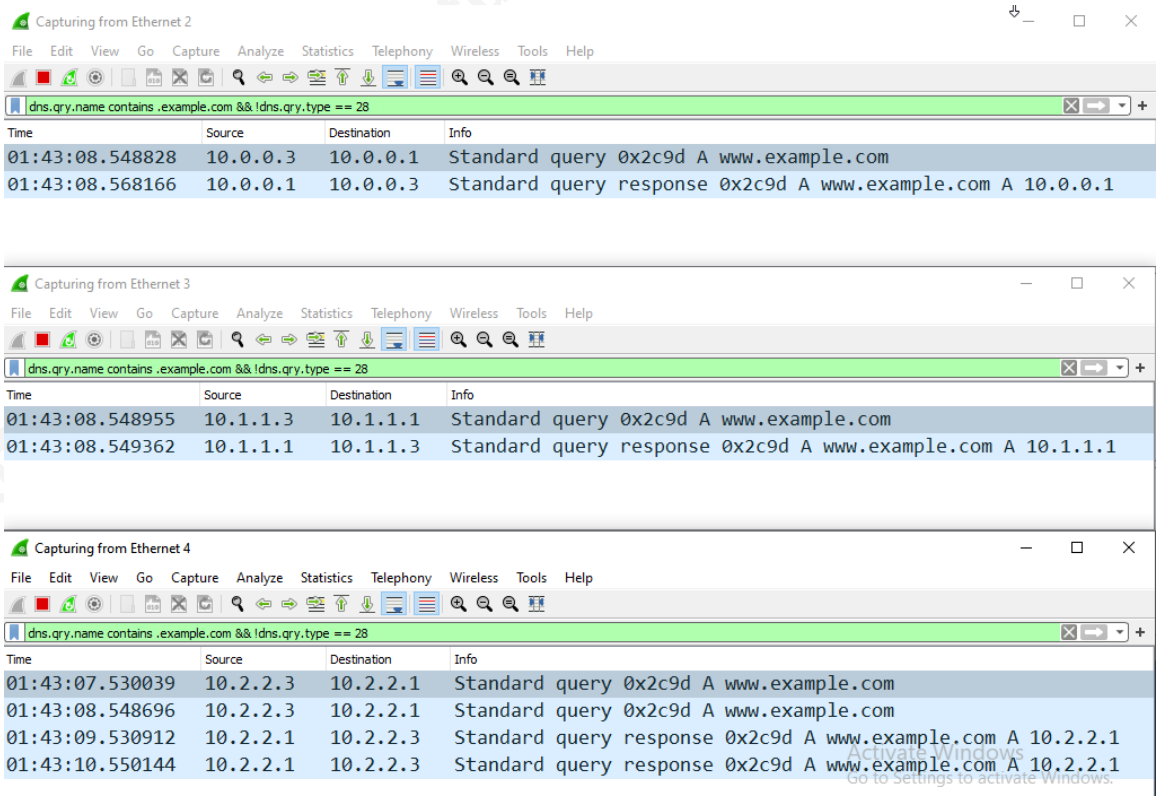

Figure 11. Ethernet 3 First Response

```
PS C:\Windows\system32> Clear-DNSClientCache
PS C:\Windows\system32> Resolve-DNSName www.example.com |FL


Name       : www.example.com
Type       : A
TTL        : 86400
DataLength : 4
Section    : Answer
IPAddress  : 10.1.1.1
```

Figure 12. Ethernet 3 Accepted Response

```
PS C:\Windows\system32> route print
===========================================================================
Interface List
  9...08 00 27 ad c2 78 ......Intel(R) PRO/1000 MT Desktop Adapter #4
  5...08 00 27 64 0f 46 ......Intel(R) PRO/1000 MT Desktop Adapter #2
  8...08 00 27 6f 07 28 ......Intel(R) PRO/1000 MT Desktop Adapter #3
  1...........................Software Loopback Interface 1
===========================================================================
```

Figure 13. Interface Binding Order (Highest Priority on Top)

Some test captures exhibited DNS query order anomalies. For example, on test 2.1.a, the DNS query order should have been Ethernet 4, then Ethernet 2, and finally Ethernet 3 (Figure 13), but the capture showed that the query went out Ethernet 4, then Ethernet 3, and finally Ethernet 2 (Figure 14). Since the queries were either simultaneous or one micro-second apart, the theory is that this was a race condition based on how Wireshark hooks into the network stack. Further investigation and testing are needed to confirm.

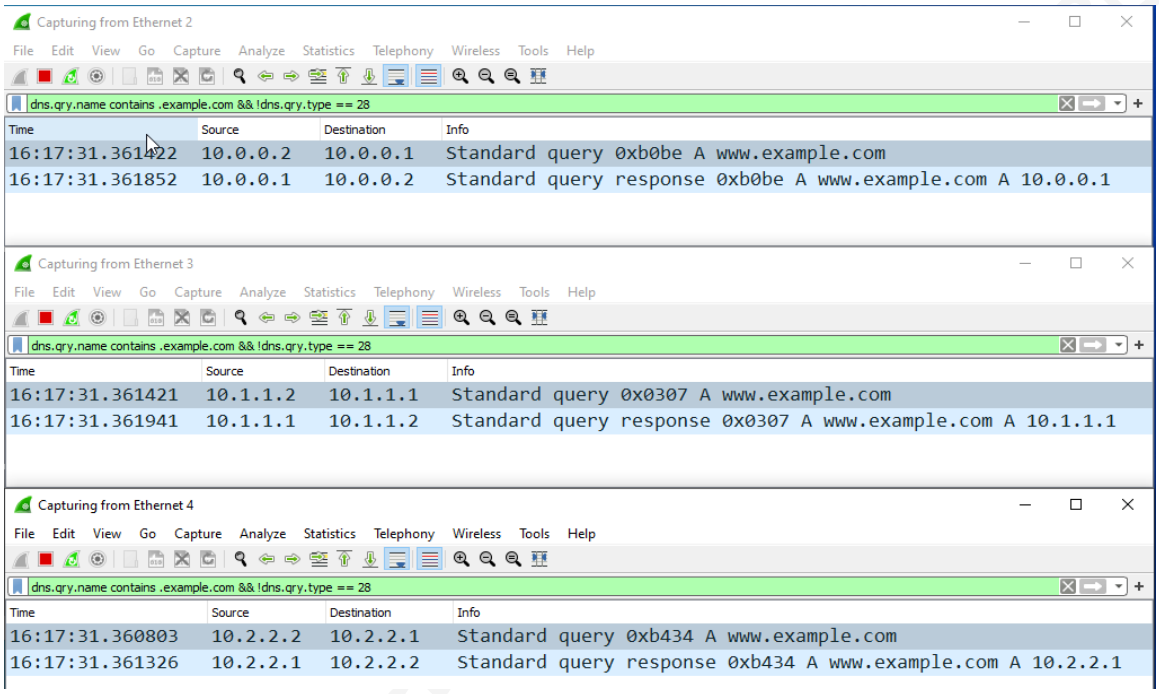

Figure 14. Test 2.1.a DNS Query Order Anomaly

3.3 Name Resolution Policy Table (NRPT) Rules Tests– Ethernet 4 Preferred Adapter

Purpose: To determine if NRPT rules provide a reliable mechanism for controlling DNS Query/Acceptance behavior.

Control: Perform a DNS query for 'www.example.com' with no NRPT rules (the default state).

Hypothesis: The DNS servers specified in the NRPT rules should be used for DNS queries while ignoring the DNS servers configured on the network adapters and the network binding order.

Tests:

3.1. Control – No NRPT Rules Present (Default State): *With no NRPT rules present, perform a DNS query for hostname www.example.com and monitor the DNS query server, the query order, and which DNS answer is accepted.*

3.2. (Ethernet 3) Example.com Only Rule: *Create an NRPT rule that specifies the Ethernet 3 DNS server, 10.1.1.1, for the example.com domain name lookups without affecting any other lookups.*

3.2.1. *Perform a DNS query for hostname www.example.com; then monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.2.2. *Perform a DNS query for hostname www.example2.com; then monitor the queried DNS server and the query order.*

3.2.3. *Perform a DNS query for hostname ftp.example.com; then monitor the queried DNS server and the query order.*

3.3. (Ethernet 3) Example.com and (Ethernet 2) Default Rule: *Leaving the example.com NRPT rule in place, create an additional NRPT rule that specifies the Ethernet 2 DNS server, 10.0.0.1, be used for all other domain lookups.*

3.3.1. *Perform a DNS query for hostname www.example.com; then monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.3.2. *Perform a DNS query for hostname www.example2.com; then monitor the queried DNS server and the query order.*

3.3.3. *Perform a DNS query for hostname ftp.example.com; then monitor the queried DNS server and the query order.*

3.4. (Ethernet 3) No DNS Server Response: *With both NRPT rules in place, disable the DNS service on DNS server 10.1.1.1, the example.com-specified lookup server, but keep the server online as it must be network reachable for this test.*

3.4.1. *Perform a DNS query for hostname www.example.com; then monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.2. *Perform a DNS query for hostname www.example.com using the 10.2.2.1 DNS server by specifying the 10.2.2.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.3. *Remove all NRPT rules and perform a DNS query for hostname www.example.com. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.4. *With no NRPT rules in place, perform a DNS query for hostname www.example.com using the 10.0.0.1 DNS server by specifying the 10.0.0.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

Test Summary: Add NRPT rules that specify which DNS Servers should be used for specific domains and default lookups, then perform a series of DNS queries to check NRPT rule enforcement.

Results:

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
3.1.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,2,3	Ethernet 4,2,3
3.2.1.a	www.example.com	Eth 3: 10.1.1.1	Ethernet 3	Ethernet 3
3.2.2.a	www.example2.com	Server Failure	Ethernet 4,2,3	Ethernet 2,4,3
3.2.3.a	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.3.1.a	www.example.com	10.1.1.1	Ethernet 3	Ethernet 3
3.3.2.a	www.example2.com	Server Failure	Ethernet 2	Ethernet 2
3.3.3.a	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.4.1.a	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.2.a	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.3.a	www.example.com	10.2.2.1	Ethernet 4,2,3	Ethernet 4,3,2
3.4.4.a	www.example.com	10.0.0.1	Ethernet 2	Ethernet 2

Table 6. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Preferred Adapter (SMHNR ON)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
3.1.b	www.example.com	Eth 4: 10.2.2.1	Ethernet 4	Ethernet 4
3.2.1.b	www.example.com	Eth 3: 10.1.1.1	Ethernet 3	Ethernet 3
3.2.2.b	www.example2.com	Server Failure	Ethernet 4,2,3	Ethernet 4,3,2
3.2.3.b	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.3.1.b	www.example.com	10.1.1.1	Ethernet 3	Ethernet 3
3.3.2.b	www.example2.com	Server Failure	Ethernet 2	Ethernet 2
3.3.3.b	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.4.1.b	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.2.b	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.3.b	www.example.com	10.2.2.1	Ethernet 4	Ethernet 4
3.4.4.b	www.example.com	10.0.0.1	Ethernet 2	Ethernet 2

Table 7. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Preferred Adapter (SMHNR OFF)

Results Summary: In all NRPT tests (3.1 to 3.4), the NRPT rules took effect and controlled which DNS servers were used for the queries, regardless of the network adapter configuration or the binding order. This effect occurred whether SMHNR was on or off. Test 3.2 provided evidence that a domain-based NRPT Rule (Figure 15) could force using a specific DNS server that was different from the server configured on the preferred adapter, Ethernet 4 (Figure 16).

```
PS C:\Windows\system32> Add-DnsClientNrptRule -namespace '.example.com' -dis
playName 'Example' -nameServers 10.1.1.1
PS C:\Windows\system32> Get-DnsClientNrptRule |FT displayName,nameSpace,name
Servers

displayName nameSpace      NameServers
----------- ---------      -----------
Example     {.example.com} 10.1.1.1
```

Figure 15. Example.com NRPT Rule

```
PS C:\Windows\system32> route print
===========================================================================
Interface List
  9...08 00 27 ad c2 78 ......Intel(R) PRO/1000 MT Desktop Adapter #4
  5...08 00 27 64 0f 46 ......Intel(R) PRO/1000 MT Desktop Adapter #2
  8...08 00 27 6f 07 28 ......Intel(R) PRO/1000 MT Desktop Adapter #3
  1...........................Software Loopback Interface 1
===========================================================================
```

Figure 16. Ethernet 4 – Preferred Adapter

As a result, the DNS query only went out the interface, Ethernet 3, which routed to the DNS server specified in the NRPT Rule (Figure 17).

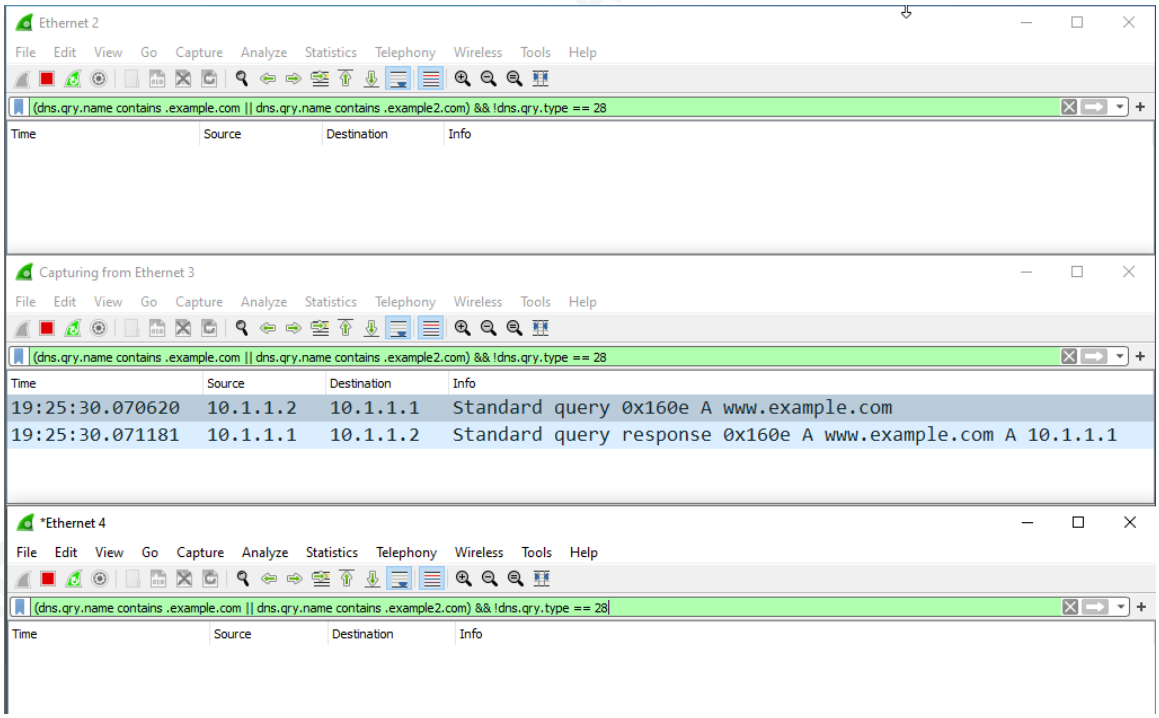

Figure 17. Ethernet 3 – DNS Query

DNS queries for domains without a corresponding NRPT Rule exhibited standard DNS behavior (Figure 18).

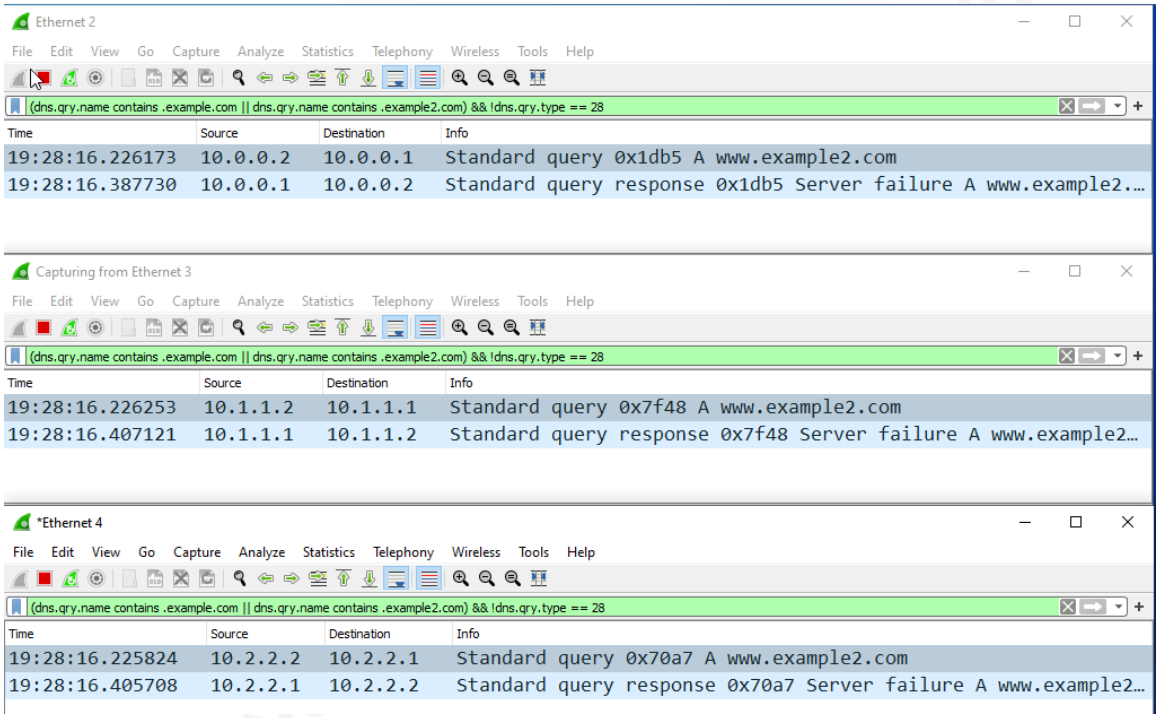

Figure 18. www.example2.com DNS Query – No NRPT Rule

Test 3.3 further illustrated that adding a default NRPT Rule changed the DNS server used for all DNS queries that did not already have a rule (Figure 19) and successfully overrode the network adapter's DNS server configuration (Figure 20).

```
PS C:\Windows\system32> Add-DnsClientNrptRule -namespace '.' -displayName 'De
fault' -nameServers 10.0.0.1
PS C:\Windows\system32> Get-DnsClientNrptRule |FT displayName,nameSpace,nameS
ervers

displayName nameSpace      NameServers
----------- ---------      -----------
Example     {.example.com} 10.1.1.1
Default     {.}            10.0.0.1
```

Figure 19. Default and Example.com NRPT Rules

Robert Upchurch, robert.upchurch@student.sans.edu

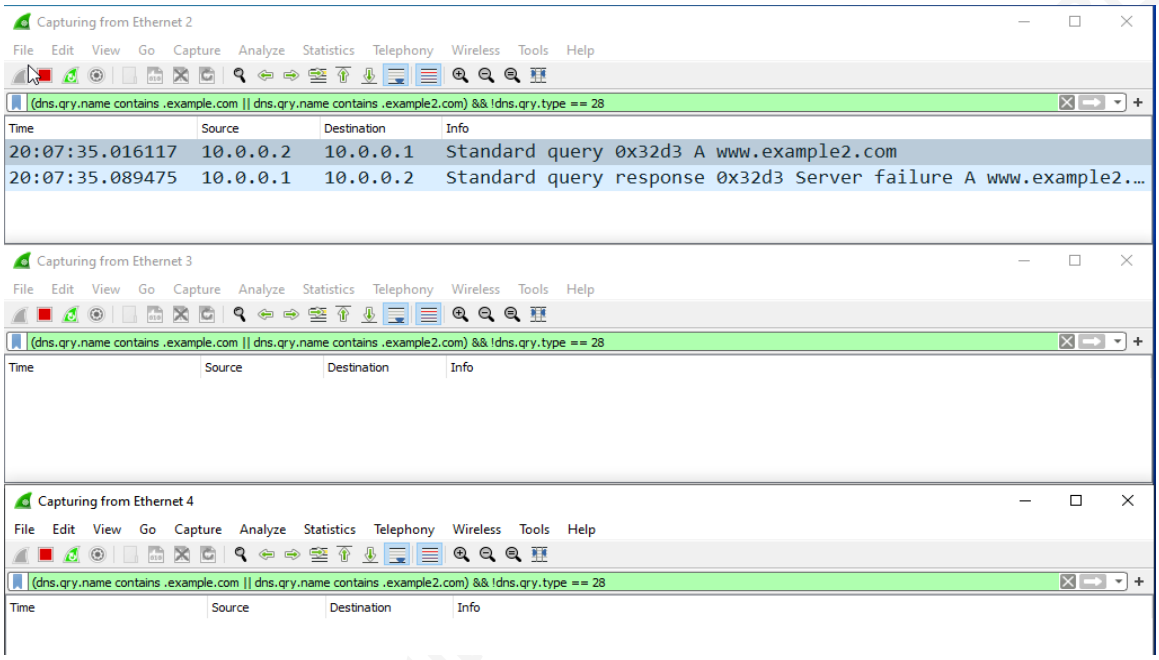

Figure 20. www.example2.com DNS Query using Default NRPT Rule

Note, however, that the query for www.example.com still used the Ethernet 3 DNS server specified in the NRPT Rule (Figure 21).

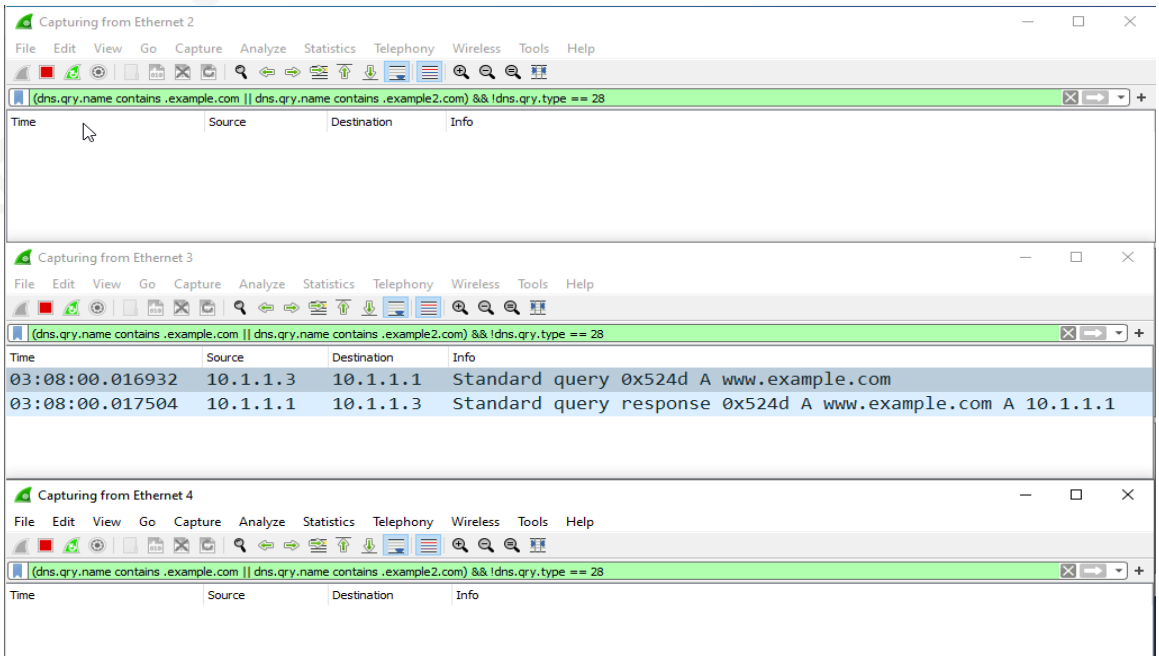

Figure 21. www.example.com DNS Query using Example.com NRPT Rule

NRPT rule effects persisted for negative DNS responses as well. In the case that network adapters had DNS servers configured on them, no queries would be sent to them unless they were present in the NRPT rule, even if all the DNS servers specified in the NRPT rule were unavailable (Figure 22).

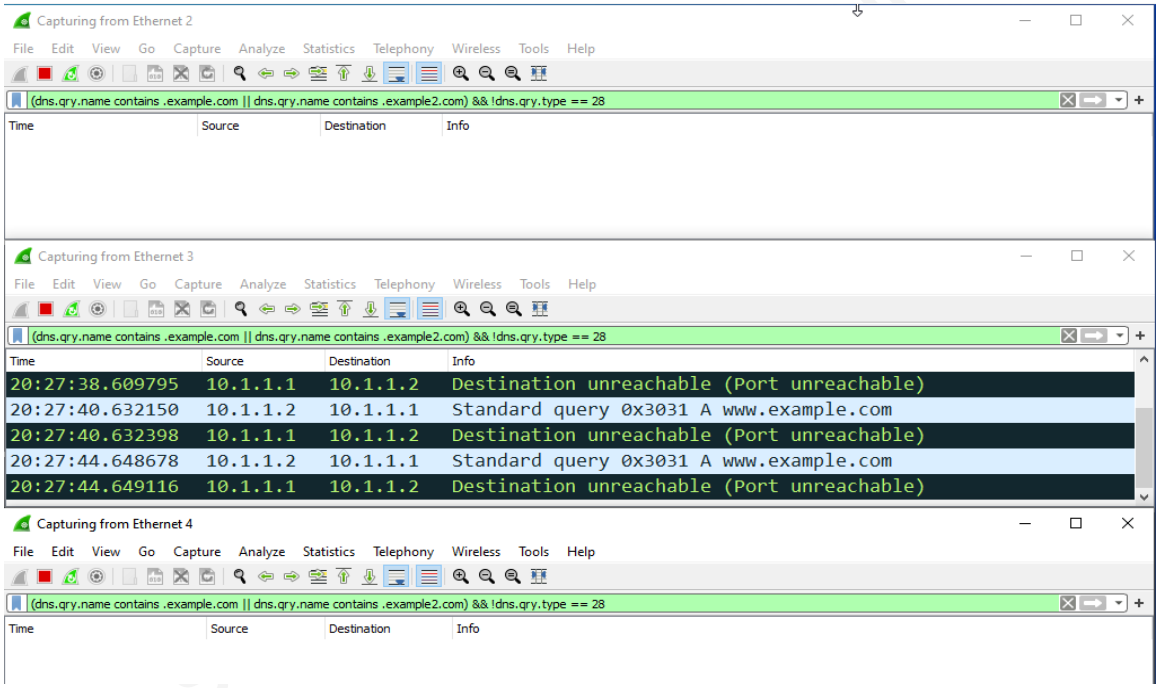

Figure 22. Ethernet 3 DNS Server Unavailable

3.4 Tests Summary

1. Interface Priority
   1.1. Control – Default Interface Metrics: *With all network adapters set to their default interface metric of 25, perform a DNS query for hostname www.example.com; then, monitor for the DNS query server, the query order, and the accepted DNS answer.*
   1.2. Ethernet 2 – Lowest Interface Metric: *Change the Ethernet 2 interface metric to 20, keeping all other interfaces at their default settings. Perform a DNS query for hostname www.example.com; then, monitor for the DNS query server, the query order, and the accepted DNS answer.*
   1.3. Ethernet 3 – Lowest Interface Metric: *Change the Ethernet 3 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for*

hostname www.example.com; then, monitor for the DNS query server, the query order, and the accepted DNS answer.

1.4. Ethernet 4 – Lowest Interface Metric: *Change the Ethernet 4 interface metric to 20, keeping all other interfaces at their defaults. Perform a DNS query for hostname www.example.com; then, monitor for the DNS query server, the query order, and the accepted DNS answer.*

2. DNS Response – Ethernet 4 Configured as the Preferred Adapter
   2.1. Control – Positive DNS Server Response: *Perform a DNS query for hostname www.example.com; then, monitor for the DNS query server, the query order, and the accepted DNS answer.*
   2.2. NXDomain (No Such Name): *Perform a DNS query for hostname ftp.example.com, which does not have a name record; then, monitor for the DNS query server and query order.*
   2.3. (Ethernet 4) 2 Second Delay DNS Server Response: *Configure the preferred adapter's DNS server (Ethernet 4) to simulate a two-second delay. Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*
   2.4. (Ethernet 4) No DNS Server Response: *Disable the DNS service on the preferred adapter's DNS server (Ethernet 4) but keep the server online as it must be network reachable for this test. Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*
3. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Configured as the Preferred Adapter
   3.1. Control – No NRPT Rules Present (Default State): *With no NRPT rules present, perform a DNS query for hostname www.example.com and monitor the DNS server queried, the query order, and which DNS answer is accepted.*
   3.2. (Ethernet 3) Example.com Only Rule: *Create an NRPT rule that specifies the Ethernet 3 DNS server, 10.1.1.1, for the example.com domain name lookups without affecting any other lookups.*

3.2.1. *Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.2.2. *Perform a DNS query for hostname www.example2.com; then, monitor the queried DNS server and the query order.*

3.2.3. *Perform a DNS query for hostname ftp.example.com; then, monitor the queried DNS server and the query order.*

3.3. (Ethernet 3) Example.com and (Ethernet 2) Default Rule: *Leaving the example.com NRPT rule in place, create an additional NRPT rule that specifies the Ethernet 2 DNS server, 10.0.0.1, be used for all other domain lookups.*

3.3.1. *Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.3.2. *Perform a DNS query for hostname www.example2.com; then, monitor the queried DNS server and the query order.*

3.3.3. *Perform a DNS query for hostname ftp.example.com; then, monitor the queried DNS server and the query order.*

3.4. (Ethernet 3) No DNS Server Response: *With both NRPT rules in place, disable the DNS service on DNS server 10.1.1.1, the example.com-specified lookup server, but keep the server online as it must be network reachable for this test.*

3.4.1. *Perform a DNS query for hostname www.example.com; then, monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.2. *Perform a DNS query for hostname www.example.com using the 10.2.2.1 DNS server by specifying the 10.2.2.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.3. *Remove all NRPT rules and perform a DNS query for hostname www.example.com. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.4.4. *With no NRPT rules in place, perform a DNS query for hostname www.example.com using the 10.0.0.1 DNS server by specifying the 10.0.0.1 lookup server as part of the lookup command. Monitor the queried DNS server, the query order, and the accepted DNS answer.*

3.5 Results Summary

Test #	Interface Metric (Ethernet 2,3,4)	Preferred Adapter (Ethernet 2,3,4)	DNS Query Order (Ethernet 2,3,4)	Accepted DNS Response (Ethernet 2,3,4)
1.1.a	25,25,25	Ethernet 2	Ethernet 2,3,4	Ethernet 2
1.2.a	20,25.25	Ethernet 2	Ethernet 2,3,4	Ethernet 2
1.3.a	25,20,25	Ethernet 3	Ethernet 3,2,4	Ethernet 3
1.4.a	25,25,20	Ethernet 4	Ethernet 4,2,3	Ethernet 4

Table 8. Interface Priority Tests (SMHNR ON)

Test #	Interface Metric (Ethernet 2,3,4)	Preferred Adapter (Ethernet 2/3/4)	DNS Query Order (Ethernet 2,3,4)	Accepted DNS Response (Ethernet 2/3/4)
1.1.b	25,25,25	Ethernet 2	Ethernet 2	Ethernet 2
1.2.b	20,25.25	Ethernet 2	Ethernet 2	Ethernet 2
1.3.b	25,20,25	Ethernet 3	Ethernet 3	Ethernet 3
1.4.b	25,25,20	Ethernet 4	Ethernet 4	Ethernet 4

Table 9. Interface Priority Tests (SMHNR OFF)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
2.1.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,3,2	Ethernet 4,2,3
2.2.a	ftp.example.com	Does Not Exist	Ethernet 4,2,3	Ethernet 4,2,3
2.3.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,2,3	Ethernet 3,2,4
2.4.a	www.example.com	10.0.0.1	Ethernet 4,2,3	Ethernet 3,2

Table 10. DNS Response Tests – Ethernet 4 Preferred Adapter (SMHNR ON)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
2.1.b	www.example.com	Eth 4: 10.2.2.1	Ethernet 4	Ethernet 4
2.2.b	ftp.example.com	Does Not Exist	Ethernet 4,2,3	Ethernet 4,2,3
2.3.b	www.example.com	Eth 3: 10.1.1.1	Ethernet 4,2,3	Ethernet 3,2,4
2.4.b	www.example.com	10.1.1.1	Ethernet 4,2,3	Ethernet 3,2

Table 11. DNS Response Tests – Ethernet 4 Preferred Adapter (SMHNR OFF)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
3.1.a	www.example.com	Eth 4: 10.2.2.1	Ethernet 4,2,3	Ethernet 4,2,3
3.2.1.a	www.example.com	Eth 3: 10.1.1.1	Ethernet 3	Ethernet 3
3.2.2.a	www.example2.com	Server Failure	Ethernet 4,2,3	Ethernet 2,4,3
3.2.3.a	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.3.1.a	www.example.com	10.1.1.1	Ethernet 3	Ethernet 3
3.3.2.a	www.example2.com	Server Failure	Ethernet 2	Ethernet 2
3.3.3.a	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.4.1.a	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.2.a	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.3.a	www.example.com	10.2.2.1	Ethernet 4,2,3	Ethernet 4,3,2
3.4.4.a	www.example.com	10.0.0.1	Ethernet 2	Ethernet 2

Table 12. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Preferred Adapter (SMHNR ON)

Test #	DNS Query	DNS Response (Ethernet 2/3/4: Answer)	DNS Query Order (Ethernet 2,3,4)	DNS Response Order (Ethernet 2,3,4)
3.1.b	www.example.com	Eth 4: 10.2.2.1	Ethernet 4	Ethernet 4
3.2.1.b	www.example.com	Eth 3: 10.1.1.1	Ethernet 3	Ethernet 3
3.2.2.b	www.example2.com	Server Failure	Ethernet 4,2,3	Ethernet 4,3,2
3.2.3.b	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.3.1.b	www.example.com	10.1.1.1	Ethernet 3	Ethernet 3
3.3.2.b	www.example2.com	Server Failure	Ethernet 2	Ethernet 2
3.3.3.b	ftp.example.com	Does Not Exist	Ethernet 3	Ethernet 3
3.4.1.b	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.2.b	www.example.com	Timeout	Ethernet 3	Ethernet 3
3.4.3.b	www.example.com	10.2.2.1	Ethernet 4	Ethernet 4
3.4.4.b	www.example.com	10.0.0.1	Ethernet 2	Ethernet 2

Table 13. Name Resolution Policy Table (NRPT) Rules – Ethernet 4 Preferred Adapter (SMHNR OFF)

4. Recommendations

Name Resolution Policy Table rules can be used to configure domain-specific DNS servers and default DNS servers. NRPT rules provide a simple, scalable, and effective mechanism for controlling DNS Client server configurations and mitigates DNS leakage risk caused by Smart Multi-Homed Name Resolution querying an unintended DNS server. Layering NRPT rules with an encrypted transport mechanism, such as a VPN, can further mitigate DNS leakage by encrypting cleartext DNS queries. Further studies should investigate the impact that SMHNR may have on IPv6 and Link-Local Name Resolution mechanisms.

5. Conclusion

Smart Multi-Homed Name Resolution does affect DNS Client behavior; when the confidentiality of DNS Client queries is critical to the privacy, security, and safety of a user or organization, its effects may warrant consideration. DNS leakage is possible regardless of whether SMHNR is on or off. For instance, DNS queries are sent cleartext by default, based on the network binding order. Additionally, public networks can

dynamically configure the DNS Client server settings. However, with SMHNR on, the likelihood of leakage increases since initial queries are sent out of all interfaces with gateways and DNS servers configured; instead of using the preferred interface.

Statically setting all DNS server settings to the preferred DNS server or restricting DNS traffic with the Windows firewall are alternative native methods for controlling DNS Client query behavior, but these are cumbersome and not granular. The ability to override all other DNS server settings and specify DNS servers based on the Fully Qualified Domain Name (FQDN), Top Level Domains (TLD), all domains, and everywhere in between allows for simplicity, agility, and scalability.

Since DNS leakage can occur regardless of SMHNR being on or off, it is best to utilize mechanisms that can help address the issue as a whole. While some external tools and services can address the issue, Name Resolution Policy Table Rules and encrypted VPN tunnels may be the right choice if the tools are limited to the native operating system. Another benefit is that they work across the various editions of Windows, such as Windows 10 Professional edition and Windows 10 Home edition.

References

"Configure The Order". (2020, August 07). *Configure the Order of Network Interfaces*. Retrieved from docs.microsoft.com: https://docs.microsoft.com/en-us/windows-server/networking/technologies/network-subsystem/net-sub-interface-metric

"DNS Leaks". (2017, December 01). *DNS Leaks (Causes & Fixes)*. Retrieved from thebestvpn.com: https://thebestvpn.com/dns-leaks-causes-fixes/

"DNS Processes". (2013, March 04). *DNS Processes and Interactions*. Retrieved from docs.microsoft.com: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/dd197552(v=ws.10)?redirectedfrom=MSDN

"Turn Off". (2021, February 13). *Turn Off Smart Multi-Homed Name Resolution*. Retrieved from ADMX.help: https://admx.help/?Category=Windows_10_2016&Policy=Microsoft.Policies.DNSClient::DNS_SmartMultiHomedNameResolution

"Windows 10". (2015, September 03). *Windows 10 DNS Resolution Via VPN Connection Not Working*. Retrieved from Superuser: https://superuser.com/questions/966832/windows-10-dns-resolution-via-vpn-connection-not-working

"Windows 8". (2013, October 10). *Windows 8 and Windows 8.1 New Group Policy Settings*. Retrieved from docs.microsoft.com: https://docs.microsoft.com/en-us/archive/blogs/configmgrdogs/windows-8-and-windows-8-1-new-group-policy-settings

Robert Upchurch, robert.upchurch@student.sans.edu