Importance of DNS Suffixes and NetBIOS

Priasoft

DNS Suffixes?

What are DNS Suffixes, and why are they important?

DNS Suffixes are text that are appended to a host name in order to query DNS for an IP address. DNS works by use of “Domains”, equitable to namespaces and usually are a textual value that may or may not be “dotted” with other domains. “Support.microsoft.com” could be considers a domain or namespace for which there are likely many web servers that can respond to requests to that domain. There could be a server named SUPREDWA.support.microsoft.com, for example. The DNS suffix in this case is the domain “support.microsoft.com”.

When an IP address is needed for a host name, DNS can only respond based on hosts that it knows about based on domains. DNS does not currently employ a “null” domain that can contain just server names. As such, if the IP address of a server named “Server1” is needed, more detail must be added to that name before querying DNS. A suffix can be appended to that name so that the DNS sever can look at the records of the domain, looking for “Server1”. A client host can be configured with multiple DNS suffixes so that there is a “best chance” of discovery for a host name.

NetBIOS?

NetBIOS is an older Microsoft technology from a time before popularity of DNS. WINS, for those who remember, was the Microsoft service that kept a table of names (NetBIOS names) for which IP address info could be returned. However, unlike DNS, WINS had no ability to store a hierarchy of domains with any structure and was a proprietary implementation that could not easily replicate to other, non-Microsoft name servers at the time. DNS was elevated by the growth and popularization of the WWW and the Internet. DNS became the preferred name-to-IP resolution service and mapped more easily to the hierarchical nature of Active Directory.

NetBIOS names have strict rules on characters, length, and uniqueness. In a specific “domain”, a name was required to be unique. Such is also true in DNS, but DNS proves to be more flexible since sub-domains can be used to broaden the scope of a root domain. If there was a need to have 2 servers with the same name, NetBIOS would require those hosts to be members of separate and distinct Domains (an easy reference to the old NT4 Domain). Multiple sub-domains can exist in DNS at multiple levels and as long as the fully qualified name (also known as FQDN) is unique, such is acceptable. A single DNS server can provide this while in contrast, providing the same using only NetBIOS would require a separate Domain server (WINS) for each different domain. This proved to have limits for scale and flexibility.

Given the age and apparent deprecation of NetBIOS, the question may be asked here as to why the topic even exists. The fact is that many core Microsoft APIs, programs, and code executions still make calls to servers using NetBIOS names. When the further inclusion of Microsoft's RPC technology is added, the use of NetBIOS is shown to be more important.

Since the nature of NetBIOS and WINS are from a flat database view, there was no need nor implementation of any qualifying text to expand the namespaces. NetBIOS names were simply that: a name. DNS, thru the use of suffixes, can serve to provide IP addresses for NetBIOS names. However, the DNS database must obviously have the name in order for it to resolve. The extra precaution then is to either ensure that the NetBIOS name of any Microsoft host matches its DNS hostname. In cases where the NetBIOS name of a host is different than its DNS hostname, the NetBIOS name should be added to DNS for the appropriate domain.

Name Resolution Importance

Given the above introductions, it should be clear that there can and are many cases where an application, or even the Windows OS, can attempt to make a call to a remote host or server by "name only" (also known as a 'Single-Label Name'). MAPI and many RPC based protocols work this way, or have the potential. Having proper DNS suffixes is important to prevent intermittent cases of failure to connect. With this, the following will address the different context under which suffixes are important and should be employed.

Migration Hosts

All computers that will be used to host the Priasoft migration tools should have DNS suffixes of all source and target domains, if not at a minimum the domains for which Exchange servers and user account objects are hosted. The recommendation to have suffixes of all domains comes from the likely case that in a more complicated domain architecture – one where there may be a Root domain and one or more child domains – it is common for primary servers (DCs and GCs) or the server(s) that support a Windows Trust to exist in one of the domains only. RPC calls made will often attempt to communicate with the PDC emulator of a domain or forest in order to secure the communication channel, especially if that RPC call attempts to access a resource across a Forest boundary.

The Priasoft tools will use several network dependent APIs and technologies that inherently depend upon proper working name resolution, such as: MAPI, LDAP, WinRM, HTTP, RPC, Windows File and Print, etc. In addition, using only IP addresses in this modern time often will not work for secured communications such as Kerberos and SSL. Both technologies leverage Certificates to ensure that the host at a particular IP address matches the name originally queried. Attempting to make a secure connection by IP address will cause a certificate issue since the IP address will not be listed on the certificate of the target host.

There is then a temptation to try to use a "hosts" file to facilitate name resolution; such should be avoided at all costs. In many cases "hosts" is not sufficient as it cannot describe domain level information and would then require the use of LMHosts (the NetBIOS version of 'hosts') and even then service records available in DNS (like MX, SRV, and others) is not available using the "hosts" file. It is best, and supported, to use DNS properly.

Exchange Servers

Exchange Servers may also need suffixes of a remote environment, depending up the context of use and migration style. If Linked Mailboxes are to be used – a case where the mailbox resides in one forest, while the authentication and user account live in another – Exchange servers will attempt to validate access by querying details from the remote forest. If an Exchange server attempts to access a remote DC or other server by name only, Windows will append the DNS suffix of the local domain before sending to DNS. It is unlikely that the local DNS would have records of remote Exchange servers. Such is also important to mention here that one should NOT add static records ('A' records or 'CNAME' records) to a DNS server for hosts that live in another domain. While the DNS server would be able to return an IP address, the FQDN would not match the remote host's certificate and would then not be able to make the connection.

Hub transport servers, which handle mail-flow in and out of the Exchange system, will use DNS to look up MX records for cases where mail must be forwarded out of the organization. If such a query fails, so will mail flow from that server. Even more subtle is a case where the Hub Transport IS able to resolve an MX record, but does so from the Public DNS; in such a case the IP address would likely be a value that is addressable on the Internet and may be unexpected or inappropriate, especially if the intent is to simply relay mail from one Exchange environment to another.

Exchange 2010 and later employ several services at which developers, script writers, and applications can interact. Services like Remote PowerShell and Exchange Web Services (EWS) cause actions to take place on the Exchange server and not on the host making the call. Many operations, like creating a mailbox, can cause the Exchange server to validate settings supplied to the call and such may trigger name resolution. This means that the name resolution is occurring on the Exchange server and therefore needs to be able to properly resolve potential single-label names.

Domain Controllers

In environments where a Windows Trust is required, name resolution is just as important. The ability of a Trust Server in one domain to validate authentication requests from a remote domain rely on name resolution and are still built up RPC and thus can end up using NetBIOS names.

The use of Linked Mailboxes show this need quickly since it is a requirement that a Windows Trust be employed to support the option. Not only will calls to create such a mailbox cause name resolution to occur, but later when a logon attempt is made to a Linked Mailbox, the authentication process will trigger cross-forest name resolution as well.

In such cases, it can then be important for DCs to also have DNS suffixes of remote domains to facility name resolution. It is not enough to have DNS services running on a DC. A Windows Domain Controller is still just a windows host, and will use standard Windows APIs for name resolution, the same as any other host.

End User Workstations

The above sections addressed what is typically the "target" side of a migration effort. However, care and planning for end-user workstations must also be considered. If a user's workstation is a member of a domain or forest that is different than the domain hosting the Exchange server hosting that user's mailbox, it becomes important for DNS suffixes to exist on the user's computer for the "target" domain. When Outlook attempts to connect to the mailbox server, it very often at some point, will attempt to access a server by name only. Without the suffixes of the target environment, Outlook will fail to connect in such cases.

In a reversed context, if user workstations are a member of the "target" domain, and yet the mailboxes remain in the "source" (usually facilitated by SID history) DNS suffixes again become important. In this case, the suffixes of the source domain(s) should be added so that in any case of a single-label name being used, there are suffixes available to help with name resolution.

Source versus Target

In many migrations there is a definition of Source and Target and it relates to the direction of flow of data and settings migrated. When reviewing the above topics, all the cases above should be analyzed in reverse as there can very often be cases where Source DCs or Exchange Servers need to be able to resolve names of hosts in the Target.

DNS Suffix Order

The DNS Suffixes are used in the order in which they are stored on a system, top to bottom. In most cases, the first suffixes should be for the domains that are of the same Forest for the host. If a migration host is a member of the "Priasoft.com" Forest, and is specifically a member of the child domain "US", such produces an FQDN of "hostname.US.Priasoft.com". This host should have suffixes of "US.Priasoft.com" first, followed by "Priasoft.com" second, followed then by any other domains necessary.

This is generally good form, however that can be very specific cases where the order should be changed. If there is a case where a host name is ambiguous between 2 environments, perhaps the hostname "mail", it may be better on a migration host to list the source DNS suffix first. Doing so will ensure that calls to "mail.source.com" are resolved to an IP address in the source domain, even though there may also be a host in "US.Priasoft.com" also with the name "mail" (ex: mail.US.Priasoft.com). Having the source suffix first allows resolution to the source, and a different name can be used for the target (e.g. 'ex-svr-1.us.priasoft.com' instead of 'mail').

However, full analysis of such a case should be made since although very undesirable, it could happen that there is complete ambiguity between the domains. In such a case, there may be a requirement to change server or employ the previously avoided "hosts" file. If such a case does exist, communication with Priasoft support should be made to work thru the situation.

Conclusion

DNS is a core service to the success of any migration effort. The use of secure communications like Kerberos and SSL rely on certificates to ensure that the IP address used identifies the host as the name requested. Applications and APIs can, and still do make connections using "single-label names", names that are just a "name". DNS suffixes provide a way for a client to issue a series of queries until the name is resolved, or, if all suffixes produce no result, fail.

Suffixes are important for ALL servers and hosts between 2 environments that must share data, even if only temporary. Support cases show that name resolution issues are the #2 cause of problems. If questions exist about DNS, suffixes, or the information presented here, please contact support@priasoft.com and, schedules permitting, we may be able to have a discussion on this topic.