# Understand the Operation of DNS on ASA when FQDN Objects Are Used

## Contents



## Introduction

This document describes the operation of Domain Name System (DNS) on Cisco Adaptive Security Appliance (ASA) when FDQN objects are used.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of Cisco ASA.

### Components Used

In order to elucidate the workings of the DNS when multiple FQDNs are configured on the ASA in a simulated production environment, an ASAv with one interface facing the internet and one interface connected to a PC device hosted on the ESXi server was setup. The ASAv interim code 9.8.4(10) was used for this simulation.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

### Network Diagram

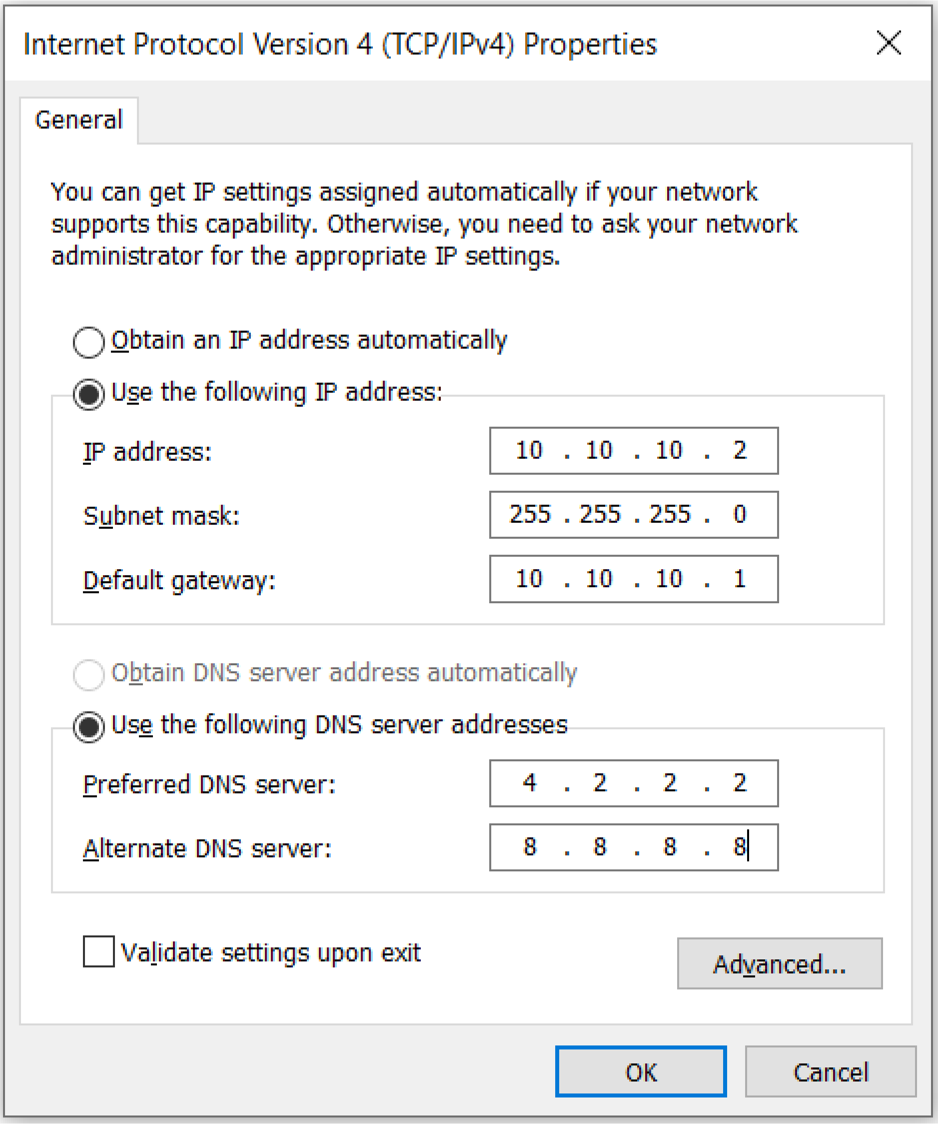
The topology setup is shown here.


## Background Information

When multiple Fully Qualified Domain Name (FQDN) objects are configured on an ASA, an end-user that tries to access any of the URLs defined in the FQDN objects would observe multiple DNS queries sent by the ASA. This document aims to provide a better understanding of why such behavior is observed.

## Configure

The client PC was configured with these IP, subnet mask, and name-servers for DNS resolution.




On the ASA, two interfaces were configured, 1 inside interface with a security level of 100 to which the PC was connected, and 1 outside interface that has connectivity to the internet.

```
ciscoasa(config-if)# sh int ip br
Interface                  IP-Address      OK? Method Status                Prot
ocol
GigabitEthernet0/0         unassigned      YES unset  administratively down down
GigabitEthernet0/1         10.197.223.9    YES DHCP   up                    up
GigabitEthernet0/2         unassigned      YES unset  administratively down down
GigabitEthernet0/3         10.10.10.1      YES manual up                    up
GigabitEthernet0/4         unassigned      YES unset  administratively down up
GigabitEthernet0/5         unassigned      YES unset  administratively down up
GigabitEthernet0/6         unassigned      YES unset  administratively down down
GigabitEthernet0/7         unassigned      YES unset  administratively down up
Internal-Control0/0        127.0.1.1       YES unset  up                    up
Internal-Data0/0           unassigned      YES unset  up                    up
Internal-Data0/1           unassigned      YES unset  up                    up
Internal-Data0/2           unassigned      YES unset  up                    up
Management0/0              unassigned      YES unset  up                    up
ciscoasa(config-if)#
```

Here Gig0/1 interface is the outside interface with an interface IP of 10.197.223.9 and the Gig0/3 interface is the inside interface with an interface IP of 10.10.10.1 and connected to the PC on the other end.

```
ciscoasa(config-if)# ping 10.197.222.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.197.222.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
ciscoasa(config-if)# ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/8/10 ms
```

Configure the DNS setup on the ASA as shown here:

```
ciscoasa(config)# sh run dns
dns domain-lookup outside
DNS server-group DefaultDNS
    name-server 4.2.2.2
ciscoasa(config)#
```

Configure 4 FQDN objects for www.facebook.com, www.google.com, www.instagram.com, and www.twitter.com.

```
ciscoasa(config)# sh run object
object network OBJ_GENERIC_ALL
 subnet 0.0.0.0 0.0.0.0
object network facebook.com
 fqdn www.facebook.com
object network twitter.com
 fqdn www.twitter.com
object network instagram.com
 fqdn www.instagram.com
object network google.com
 fqdn www.google.com
```

Set up a capture on the ASA outside interface to capture DNS traffic. Then from the client PC, try to access www.google.com from a browser.

What do you observe? Take a look at the packet capture.

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1 | 0.000000 | 10.197.223.9 | 4.2.2.2 | DNS | 76 | Standard query 0x5315 A www.f |
| 2 | 0.289078 | 4.2.2.2 | 10.197.223.9 | DNS | 364 | Standard query response 0x531 |
| 3 | 6.920002 | 10.197.223.9 | 4.2.2.2 | DNS | 77 | Standard query 0x89c3 A www.i |
| 4 | 6.965044 | 4.2.2.2 | 10.197.223.9 | DNS | 380 | Standard query response 0x89c |
| 5 | 11.959978 | 10.197.223.9 | 4.2.2.2 | DNS | 77 | Standard query 0xafb3 A www.i |
| 6 | 12.083278 | 4.2.2.2 | 10.197.223.9 | DNS | 380 | Standard query response 0xafb |
| 7 | 59.999984 | 10.197.223.9 | 4.2.2.2 | DNS | 76 | Standard query 0x9ab6 A www.f |
| 8 | 60.049268 | 4.2.2.2 | 10.197.223.9 | DNS | 364 | Standard query response 0x9ab |
| 9 | 65.039991 | 10.197.223.9 | 4.2.2.2 | DNS | 76 | Standard query 0xa89f A www.f |
| 10 | 65.089930 | 4.2.2.2 | 10.197.223.9 | DNS | 364 | Standard query response 0xa89 |
| 11 | 67.209965 | 10.197.223.9 | 4.2.2.2 | DNS | 77 | Standard query 0x66a2 A www.i |
| 12 | 67.261766 | 4.2.2.2 | 10.197.223.9 | DNS | 380 | Standard query response 0x66a |
| 13 | 72.259965 | 10.197.223.9 | 4.2.2.2 | DNS | 77 | Standard query 0x540e A www.i |
| 14 | 72.304687 | 4.2.2.2 | 10.197.223.9 | DNS | 380 | Standard query response 0x540 |
| 15 | 80.299972 | 10.197.223.9 | 4.2.2.2 | DNS | 77 | Standard query 0xf27e A www.i |
| 16 | 80.425805 | 4.2.2.2 | 10.197.223.9 | DNS | 380 | Standard query response 0xf27 |
| 17 | 84.920002 | 10.197.223.9 | 4.2.2.2 | DNS | 74 | Standard query 0xc0bb A www.g |
| 18 | 85.008498 | 4.2.2.2 | 10.197.223.9 | DNS | 338 | Standard query response 0xc0b |

Here we see that even though we tried to resolve only www.google.com, there are DNS queries sent out for all of the FQDN objects.

Now take a look at how DNS caching works for IPs on the ASA to understand why this happens.

- When www.google.com is typed in the client PCs web browser, the PC sends out a DNS query to get the URL resolved to an IP address.
- The DNS server then resolves the PCs request and returns an IP that states google.com resides at the specified location.
- The PC then initiates a TCP connection to google.com's resolved IP address. However, when the packet reaches the ASA, it does not have an ACL rule that states the specified IP is permitted or denied.
- The ASA, however, knows that it has 4 FQDN objects and that any of the FQDN objects could possibly be resolved to the concerned IP.
- Hence the ASA sends out DNS queries for all the FQDN objects as it does not know which FQDN object can resolve to the concerned IP. (This is why there are multiple DNS queries observed).
- The DNS server resolves the FQDN objects with their corresponding IP addresses. The FQDN object can get resolved to the same public IP address as was resolved by the client. Otherwise, the ASA creates a dynamic access-list entry for a different IP address than the one that the client tries to reach, hence the ASA ends up dropping the packet. For example, if the user resolved google.com to 203.0.113.1 and if the ASA resolves it to 203.0.113.2, the ASA creates a new dynamic access-list entry for 203.0.113.2 and the user are unable to access the website.
- The next time when a request arrives, that requests resolution of a particular IP, if that particular IP is stored on the ASA, it does not query all the FQDN objects again since a dynamic ACL entry would now be present.

- If a client is concerned about the large number of DNS queries sent by ASA, increase the DNS timer expiry, and provided end hosts tries to access the destination IP addresses which are there in the DNS cache. If the PC requests for an IP, not stored on the ASA DNS cache, DNS queries are sent out to resolve all the FQDN objects.

- A possible workaround for this, if you want to still reduce the number of DNS queries, would be to either reduce the  number of FQDN objects or to define the whole range of public IPs that you would resolve the FQDN to, which however defeats the purpose of an FQDN object in the first place. Cisco Firepower Threat Defense (FTD) is a better solution to handle this use case.

# Verify

In order to verify which IPs are present in the ASAs DNS cache to which each of the FQDN objects get resolved, the command **ASA# sh dns** can be used.

```
ciscoasa(config)# sh dns
Name: www.facebook.com
  Address: 157.240.192.35                                TTL 00:01:06
Name: www.google.com
  Address: 172.217.166.164                               TTL 00:04:44
Name: www.instagram.com
  Address: 157.240.16.174                                TTL 00:01:21
Name: www.twitter.com
  Address: 104.244.42.65                                 TTL 00:06:37
  Address: 104.244.42.1                                  TTL 00:05:26
```

# Related Information

[Cisco Technical Support and Downloads]