Leveraging Consul's DNS Interface

December 28, 2015 Joe Keegan Consul, DNS

Consul provides a DNS interface to query for service discovery information. While there is documentation on how to query the service data, I didn't find the documentation too informative on how your clients utilized the DNS interface. This post attempts to flesh the configuration options for leveraging the Consul DNS interface.

The basics is that, by default, the Consul agent runs a DNS server listening on port 8600. By submitting DNS requests to the Consul agent's DNS server you can get the IP address of a node running the service you are interested in. More details can be found on the DNS Interface doc page.

It's neat that Consul provides a DNS interface, in addition to the HTTP interface, but not very clear how to get your systems to query the DNS interface for service information.

The DNS Interface doc page outlines the following three options "*One option is to* ***use a custom DNS resolver library and point it at Consul****. Another option is to* ***set Consul as the DNS server for a node and provide a recursors configuration*** *so that non-Consul queries can also be resolved. The last method is to* ***forward all queries for the "consul." domain to a Consul agent from the existing DNS server****.*"

Details on the options laid out by HashiCorp are as follows.

Use a Custom DNS Resolver

This option would be to use a DNS resolver library for your language of choice. In this case your code would query the DNS interface directly. It's not clear why someone would choose to have their code utilize the DNS interface vs utilizing the HTTP interface.

Something of notes is that the Consul DNS interface makes the port information for a service available via SRV records, but you must specifically query for the SRV record type. It's very unlikely that your client will look for this record type by default and the only real practical way would be to build the logic to look for the SRV record into your code.

Without custom logic in your code you are generally limited to just the IP address info (i.e. A records) of the service you are querying for. This means if you plan to use the DNS interface then your service must run on a well known port and you client must know what port to connect to. This means that you may only run one instance of your service per IP addresses.

Use Consul DNS Server For a Node

In this option you would configure the host to send DNS queries directly to the local Consul agent's DNS server. This requires changes to both the system and Consul agent configuration.

First the system needs to be configured to send queries to the local Consul agent's DNS server. This is done by updating the resolv.conf file on the system to point to 127.0.0.1. In most cases Consul will need to be configured to run on port 53 (meaning it must be run as root) as most systems do not support having port numbers specified in the resolv.conf.

The resolv.conf should look something like:

```
search somedomain.com
nameserver 127.0.0.1
```

By default Consul will only resolve DNS queries for records within the consul top level domain. An example would be web.service.consul. To change this the client must be configured with the "recursors" config option. A list of DNS Server IP addresses that will be used to resolve records for requests outside the consul TLD are provided for the “recursors” config option.

Additionally, if you are unable to specify a port as part of your resolv.conf config you will need to configure the Consul agent to listen on port 53 for DNS queries.

Your Consul agent config should look something like this:

```
{
  "datacenter": "dc1",
  "data_dir": "/var/consul",
  "recursors" : [ "8.8.8.8" ],
  "ports" : {
    "dns" : 53
  },
  "retry_join": [ "10.0.0.100",  "10.0.1.100", "10.0.2.100" ]
}
```

In the above config any DNS query for a record that is not part of the consul TLD will be forwarded to 8.8.8.8 to be resolved.

It’s important to note that the Consul agent forwards the DNS queries on to the recursors. Meaning the system must have the ability to send DNS traffic to the IP addresses listed in the recursors config item directly.

The Consul agent will continue to be able to resolve records for records outside of the consul TLD even if the server cluster is down or unreachable.

Forward Queries for Consul TLD From a DNS Server

In this option all DNS queries are sent to a DNS server, such as a BIND/named server, and having that server forward any requests for records in the consul TLD to the DNS server run by a consul agent.

I think the best option is the have multiple BIND servers each running a Consul agent locally. Any queries received by a BIND server would be forwarded to it's local Consul agent DNS server.

Running the Consul agent on each BIND server makes the most sense to me, but it is not strictly required. If you want to have the BIND server forward a DNS request to a Consul agent running on a remote machine you will need to configure Consul on the remote machine to listen on an IP address other than 127.0.0.1, which is the default.

An example of how to configure BIND to forward queries for the consul TLD to a Consul agent can be found in the DNS Forwarding doc page.

This option provides the interesting benefit that any system can use DNS to take advantage of Consul's service discovery.

This option might not make sense if you are leveraging something like Route 53 Private Hosted Zones as your internal DNS. Unfortunately Route 53 doesn't support the ability to do any forwarding.