DoH服务器搭建调研

DNS-over-HTTPs，简称DoH，是一种加密DNS的实现方案，由IETF在2018年提出。DoH采用HTTPs报文封装DNS请求，达到DNS加密的效果，其安全性由TLS协议保证。目前，DoH还在快速发展的过程中，技术圈出现了各种各样DoH服务部署工具，他们在服务器架构、实现语言等方面各有不同，性能之间也存在差异。本文调研并使用一些DoH服务端部署工具。

下图给出了常用的DoH工具，本次调研也是围绕这些工具展开。

工具名称	项目链接	编写语言	标准GET	标准POST	JSON
dnsdist	https://www.dnsdist.org/、https://github.com/PowerDNS/pdns	Lua	√	√	X
doh-proxy	https://github.com/facebookexperimental/doh-proxy	python3	√	X	X
Coredns	https://github.com/coredns/coredns、https://coredns.io/	go	√	√	X
dns-over-https	https://github.com/m13253/dns-over-https	go	√	√	√

实验环境环境

服务器环境

- 操作系统：centos8

测试端环境

- 操作系统：win 10
- DoH客户端：dnslookup、chrome
- 捕包工具：wireshark

Dnsdist&PowerDNS

- 项目链接：https://www.dnsdist.org/guides/dns-over-https.html

PowerDNS 是一个跨平台的开源DNS服务组件，PowerDNS同时有Win32和Linux/Unix的版本。PowerDNS项目下有三个子项目，分别是：

- PowerDNS Authoritative Server
- PowerDNS Recursor
- dnsdist

其中dnsdist负责监听DNS请求并做负载均衡，转发到配置后端真正的DNS服务器中。

dnsdist is a highly DNS-, DoS- and abuse-aware loadbalancer. Its goal in life is to route traffic to the best server, delivering top performance to legitimate users while shunting or blocking abusive traffic.

dnsdist由Lua语言编写，具有很好的动态性，其配置文件可以在软件运行的过程中进行更改，可以从类似控制台的界面或HTTP API中查询其统计信息。除了支持基本的DNS协议外，dnsdist同时支持DoH、DoT、DNSCrypt等多种加密DNS协议。

搭建DoH服务器

运行以 下命令安装dnsdist

```
yum install epel-release &&
dnf install -y 'dnf-command(config-manager)' &&
dnf config-manager --set-enabled PowerTools &&
curl -o /etc/yum.repos.d/powerdns-dnsdist-15.repo https://repo.powerdns.com/repo-
files/centos-dnsdist-15.repo &&
yum install dnsdist
```

编写dnsdist配置文件`dnsdist.conf`

```
addDOHLocal('0.0.0.0:443', '/usr/local/nginx/cert/mutiply.crt',
'/usr/local/nginx/cert/mutiply.key',"/dns-query")
setACL("0.0.0.0/0")
newServer("127.0.0.1")
```

dnsdist配置DoH服务器十分方便，仅需一条命令:`addDOHLocal`。该命令使用证书和公私密钥对开启一个DoH服务器，命令参数中可用IP:port的形式指定服务器监听的IP和端口。`setACL`命令用来限定服务的IP范围，参数值为网段net/prefix。

运行服务器

```
dnsdist -C dnsdist.conf
```

验证服务器有效性

POST方式

利用dnslookup工具发送POST方式标准DoH请求，观察服务器响应

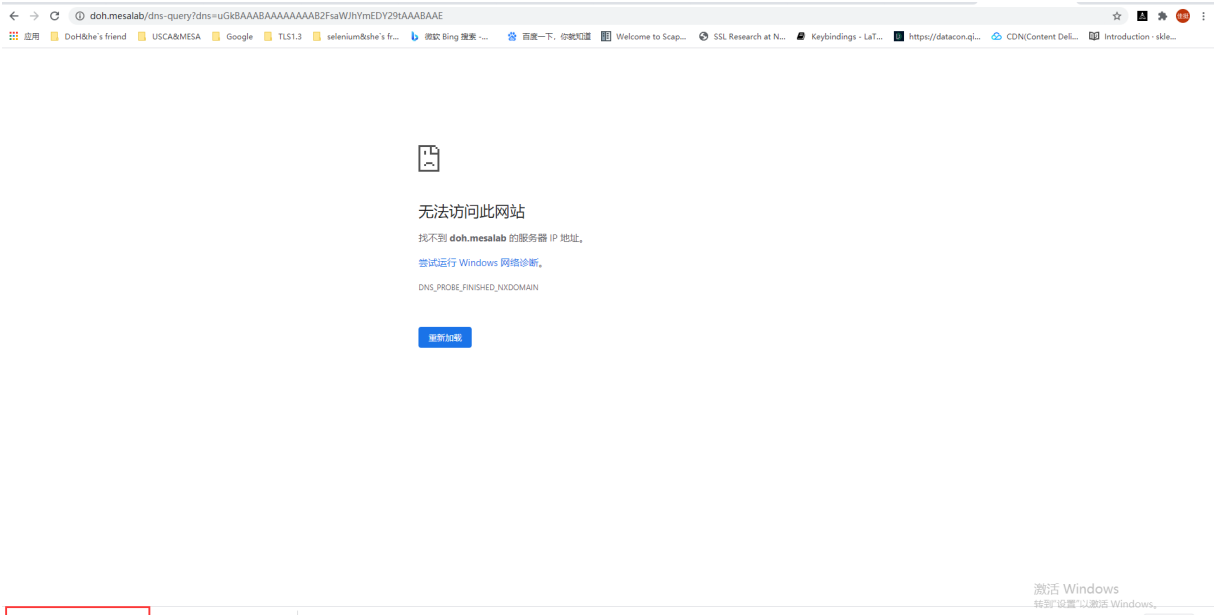
```
dnslookup.exe baidu.com https://doh.mesalab.cn/dns-query
```

成功收到应答

GET方式

在浏览器中发送GET方式标准DoH请求，观察服务器响应。

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?dns=uGkBAAABAAAAAAAAB2FsaWJhYmEDY29tAAABAAE`

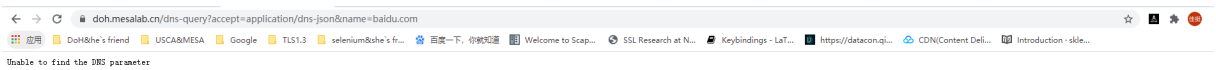


成功收到应答,服务器支持GET方式标准DoH请求。

JSON格式

在浏览器中发送json格式DoH请求，观察服务器响应

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?accept=application/dns-json&name=baidu.com`



服务器无法解析DNS参数。服务器不支持json格式的DoH请求。

基本结论

dnsdist搭建的DoH服务器支持标准格式的DoH请求,包括GET和POST两种方式，不支持json格式的请求。

doh-proxy(facebook)

- 项目链接：https://github.com/facebookexperimental/doh-proxy

没找到明确的项目介绍，目测是facebook的一个实验性功能工具。

该工具使用python3编写，已编写成python的第三方库，包含四个子工具：

- doh-proxy
- doh-httpproxy
- doh-stub
- doh-client

其中，doh-proxy可以用作搭建DoH服务器；doh-httpproxy可以用作反向代理；doh-stub做存根服务器，完成DNS和DoH数据包之间的相互转换；doh-client则是一个DoH客户端，可以发起DoH请求。

搭建DoH服务器

使用pip安装doh-proxy

```
pip3 install doh-proxy
```

运行doh-proxy

```
doh-proxy --listen-address 0.0.0.0 --port 443 --upstream-resolver 8.8.8.8 --upstream-port
53 --uri /dns-query --certfile /usr/local/nginx/cert/mutiply.crt --keyfile
/usr/local/nginx/cert/mutiply.key
```

验证服务器有效性

POST方式

利用dnslookup工具发送POST方式标准DoH请求，观察服务器响应

```
dnslookup.exe baidu.com https://doh.mesalab.cn/dns-query
```

服务器未能正常响应DoH请求，服务器存在问题。

```
C:\Users\JiaTing\Desktop\windows-amd64>dnslookup.exe baidu.com https://doh.mesalab.cn/dns-
query
dnslookup v1.4.3
2020/10/30 20:38:42 Cannot make the DNS request: couldn't initialize HTTP client or
transport, cause: couldn't do a POST request to 'https://doh.mesalab.cn:443/dns-query',
cause: Get "https://doh.mesalab.cn:443/dns-query?
dns=IMwBAAABAAAAAAAACGlwdjRvbmx5BGFycGEAAAEAAQ": net/http: request canceled (Client.Timeout
exceeded while awaiting headers)
```

观察服务器端，发现报代码错误。

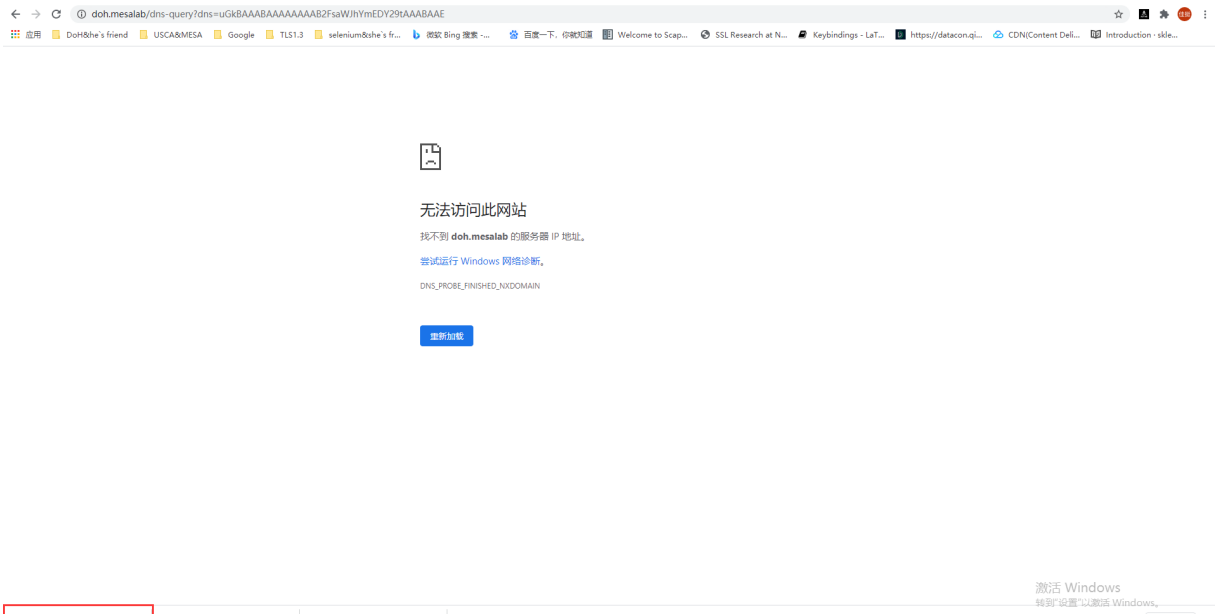
```
[root@VM-0-8-centos ~]# doh-proxy --listen-address 0.0.0.0 --port 443 --upstream-resolver
8.8.8.8 --upstream-port 53 --uri /dns-query --certfile /usr/local/nginx/cert/mutiply.crt --
keyfile /usr/local/nginx/cert/mutiply.key
2020-10-30 20:38:18,166:     INFO: Serving on <Server sockets=[<socket.socket fd=6,
family=AddressFamily.AF_INET, type=2049, proto=6, laddr=('0.0.0.0', 443)>]>
2020-10-30 20:38:31,733:     INFO: [HTTPS] 202.43.148.188 ipv4only.arpa. A IN 8396 RD
2020-10-30 20:38:31,734:     INFO: [DNS] 202.43.148.188 ipv4only.arpa. A IN 12660 RD
2020-10-30 20:38:41,735:    DEBUG: Request timed out
```

```
2020-10-30 20:38:42,806:     INFO: [DNS] 202.43.148.188 ipv4only.arpa. A IN 9079 RD
2020-10-30 20:38:42,859:     INFO: [DNS] 202.43.148.188 ipv4only.arpa. A IN 9079 QR/RD/RA
2/0/0 -1/0/0 NOERROR 53ms
2020-10-30 20:38:42,860:    ERROR: Task exception was never retrieved
future: <Task finished coro=<H2Protocol.resolve() done, defined at
/usr/local/lib/python3.6/site-packages/dohproxy/proxy.py:194>
exception=TypeError("'NoneType' object is not subscriptable",)>
Traceback (most recent call last):
  File "/usr/local/lib/python3.6/site-packages/dohproxy/proxy.py", line 203, in resolve
    self.on_answer(stream_id, dnsr=dnsr)
  File "/usr/local/lib/python3.6/site-packages/dohproxy/proxy.py", line 175, in on_answer
    clientip = self.transport.get_extra_info('peername')[0]
TypeError: 'NoneType' object is not subscriptable
```

GET方式

在浏览器中发送GET方式标准DoH请求，观察服务器响应。

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?dns=uGkBAAABAAAAAAAAB2FsaWJhYmEDY29tAAABAAE`

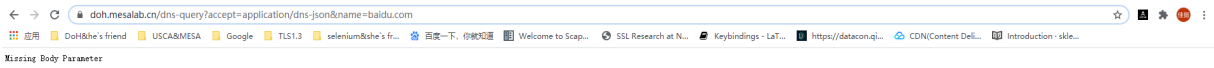


成功收到应答

JSON格式

在浏览器中发送json格式请求

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?accept=application/dns-json&name=baidu.com`



服务器无法解析DNS参数，服务器不支持json格式的DoH请求。

基本结论

doh-proxy搭建的DoH服务器仅支持标准GET方式的DoH请求；对于标准POST方式的DoH请求，服务器发生错误；同时也不支持JSON格式的DoH请求

Coredns

- 项目链接：https://github.com/coredns/coredns、https://coredns.io/

Coredns是一个DNS服务器工具，用go语言编写，特点是十分灵活，DNS的每个功能都通过一个插件的单独实现，可用根据自己的需要进行编译。其现有插件如下

Plugins

All in-tree plugins for CoreDNS.
v1.8.0

Writing Plugins | Documenting Plugins | Enabling Plugins

acl

acl enforces access control policies on source ip and prevents unauthorized access to DNS servers.
Source

any

any gives a minimal response to ANY queries.
Source

auto

auto enables serving zone data from an RFC 1035-style master file, which is automatically picked up from disk.
Source

autopath

autopath allows for server-side search path completion.
Source

azure

azure enables serving zone data from Microsoft Azure DNS service.
Source

bind

bind overrides the host to which the server should bind.
Source

bufsize

bufsize sizes EDNS0 buffer size to prevent IP fragmentation.
Source

cache

cache enables a frontend cache.
Source

cancel

cancel cancels a request's context after 5001 milliseconds.
Source

chaos

chaos allows for responding to TXT queries in the CH class.
Source

clouddns

clouddns enables serving zone data from GCP Cloud DNS.
Source

debug

debug disables the automatic recovery upon a crash so that you'll get a nice stack trace.
Source

dns64

dns64 enables DNS64 IPv6 transition mechanism.
Source

dnssec

dnssec enables on-the-fly DNSSEC signing of served data.
Source

dnstap

dnstap enables logging to dnstap.
Source

erratic

erratic a plugin useful for testing client behavior.
Source

errors

errors enables error logging.
Source

etcd

etcd enables SkyDNS service discovery from etcd.
Source

file

file enables serving zone data from an

forward

forward facilitates proxying DNS

grpc

grpc facilitates proxying DNS

RFC 1035-style master file.
Source

messages to upstream resolvers.
Source

messages to upstream resolvers via gRPC protocol.
Source

health

health enables a health check endpoint.
Source

hosts

hosts enables serving zone data from a `/etc/hosts` style file.
Source

import

import includes files or references snippets from a Corefile.
Source

k8s_external

k8s_external resolves load balancer and external IPs from outside Kubernetes clusters.
Source

kubernetes

kubernetes enables reading zone data from a Kubernetes cluster.
Source

loadbalance

loadbalance randomizes the order of A, AAAA and MX records.
Source

log

log enables query logging to standard output.
Source

loop

loop detects simple forwarding loops and halts the server.
Source

metadata

metadata enables a metadata collector.
Source

prometheus

prometheus enables Prometheus metrics.
Source

nsid

nsid adds an identifier of this server to each reply.
Source

pprof

pprof publishes runtime profiling data at endpoints under `/debug/pprof`.
Source

ready

ready enables a readiness check HTTP endpoint.
Source

reload

reload allows automatic reload of a changed Corefile.
Source

rewrite

rewrite performs internal message rewriting.
Source

root

root simply specifies the root of where to find (zone) files.
Source

route53

route53 enables serving zone data from AWS route53.
Source

secondary

secondary enables serving a zone retrieved from a primary server.
Source

sign

sign adds DNSSEC records to zone files.
Source

template

template allows for dynamic responses based on the incoming query.
Source

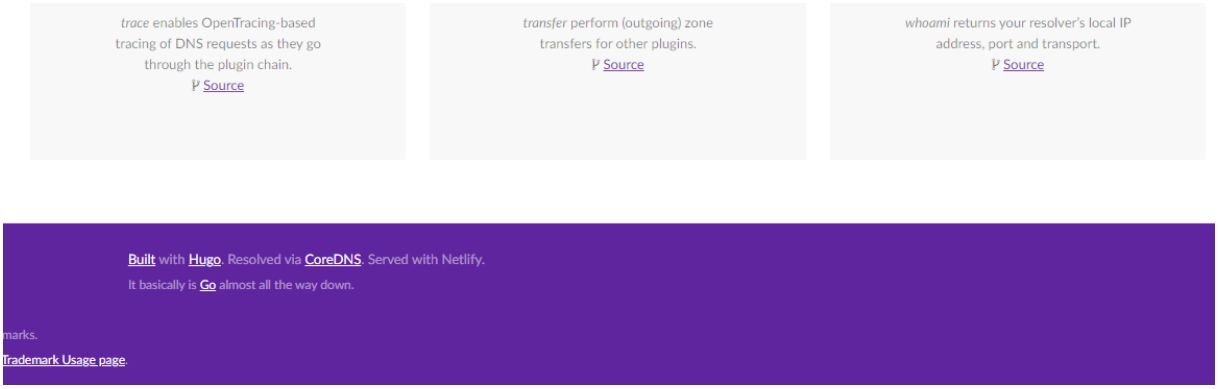
tls

tls allows you to configure the server certificates for the TLS and gRPC servers.
Source

trace

transfer

whoami



coredns本身自能用做DNS转发和加载zone文件，但配合第三方的unbound插件可以完成递归服务器功能。coredns目前能够启动DoH服务，但不可用，看具体见后续。

搭建DoH服务器

下载Coredns源码，然后编译

```
git clone https://github.com/coredns/coredns
cd coredns
make
```

coredns的默认配置文件名为`Corefile`，需要手动创建。然后在`Corefile`中输入以下内容

```
https://doh.mesalab.cn {
    whoami
    tls /usr/local/nginx/cert/mutiply.crt /usr/local/nginx/cert/mutiply.key
}
```

即可启动DoH服务。其中`doh.mesalab.cn`是和服务器IP绑定的域名，`mutiply.crt`和`mutiply.key`分别是证书和公私密钥对。

验证服务器有效方式

POST方式

利用dnslookup工具发送POST方式标准DoH请求，观察服务器响应

```
dnslookup.exe baidu.com https://doh.mesalab.cn/dns-query
```

服务器有正常DoH响应，但响应内容是拒绝服务，其原因是因为没有配置后续解析器，目前没能找到配置方法。

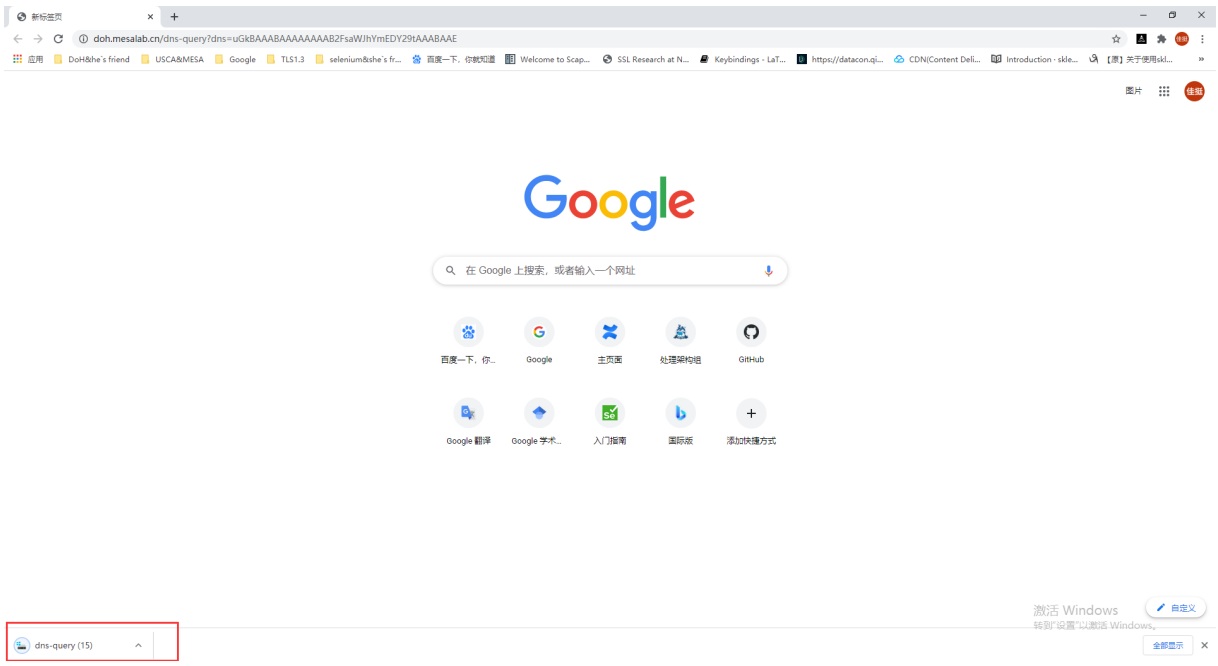
```
C:\Users\JiaTing\Desktop\doh服务器实现方式调研\windows-amd64>dnslookup.exe baidu.com
https://doh.mesalab.cn/dns-query
dnslookup v1.4.3
dnslookup result:
;; opcode: QUERY, status: REFUSED, id: 9016
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;baidu.com.      IN       A
```

GET方式

在浏览器中发送GET方式标准DoH请求，观察服务器响应。

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?dns=uGkBAAABAAAAAAAAB2FsaWJhYmEDY29tAAABAAE`

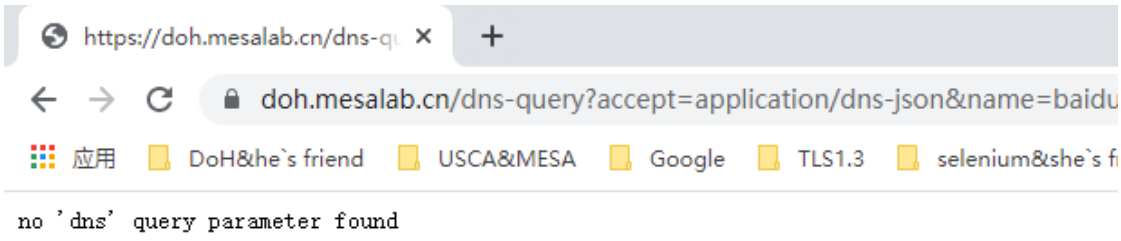


成功收到应答

JSON格式

在浏览器中发送json格式请求

在浏览器地址栏键入`https://doh.mesalab.cn/dns-query?accept=application/dns-json&name=baidu.com`



服务器无法解析DNS参数，服务器不支持json格式的DoH请求。

基本结论

Coredns搭建的DoH服务器能支持标准的GET方式和POST方式的DoH请求，不支持JSON格式的DoH请求。对于GET方式和POST方式，目前仅实现了前端，没有后续的DNS解析。

dns-over-https

- 项目链接： https://github.com/m13253/dns-over-https

dns-over-https是我们最开始采用的DoH搭建工具，其搭建过程可参考前述报告。dns-over-https用go语言编写，包含doh-client和doh-server两部分。

- doh-client可以讲系统明文DNS转换成DoH报文，作为系统级的DNS工具。
- doh-server则可以作为搭建DoH服务器的工具，十分方便，且支持标准GET、POST和JSON格式的DoH请求。