Содержание

DNS Query Processing and DNS Response Spoofing 3
Purpose 3
Service Operation Scheme 3
Configuration 4
Management 4

DNS Query Processing and DNS Response Spoofing

Purpose

The DNS response spoofing service allows modifying IP addresses returned by the DNS server for specific domain names. This enables influencing DNS server responses by redefining IP addresses in server responses for certain types of DNS queries specified in the service settings.
This service is effective for controlling client DNS queries and redirecting them to alternative IP addresses. It is used for load balancing traffic of specific resources between different servers based on IP address.
Description of DNS query and response export configuration.

Service Operation Scheme

1. The client makes a specific type of query to the DNS server (e.g., type A).
2. DPI analyzes all DNS queries and checks whether the spoofing service is assigned for this client (source IP) for a specific resource. If blocking is configured for this query, DPI simply drops the DNS query with the specific record type.
3. If the service is assigned, DPI drops the client's original DNS query and forms a DNS server response according to the rules specified in the service.
4. DPI redirects the modified response to the client. The client doesn't notice the modification and considers the response legitimate.

Supported DNS record types:

- A — IPv4 address (32-bit length);
- AAAA — IPv6 address (128-bit length);
- HTTPS — this record type is designed to provide information about available services operating over HTTPS. It allows specifying alternative endpoints, HTTP/3 support, ClientHello encryption, and non-standard TCP/UDP ports;
- MX — a TXT record containing information about mail servers handling email.

Possible actions with DNS queries:

- `ya.ru HTTPS #drop` — DPI drops the DNS query with HTTPS record type
- `ya.ru HTTPS #nxdomain` — DPI responds that the domain doesn't exist for the DNS query with HTTPS record type
- `ya.ru A #nxdomain` — DPI sends a response about the non-existence of the domain with A record type
- `mail.ru MX smtp.googlemail.com` — in this case, for a mail.ru query with MX type, the response should indicate that the mail.ru domain has a mail server at smtp.googlemail.com with a `preference` of 10.

Configuration

1. Create a text file and add DNS query processing rules, specifying: domain name, DNS record type, either an action or IP address, or domain for MX record type that will be included in the response for this domain. Wildcard * is supported for domains.

   ```
   vi test.txt
   google.com A 192.0.2.1
   test.ru A #nxdomain
   example.com AAAA 2001:db8:85a3::8a2e:370:7334
   ya.ru HTTPS #drop
   *.fb.com A 203.0.113.5
   mail.ru MX smtp.googlemail.com
   ```

2. Convert the text file to a binary format understandable by DPI using the `dns2dic` utility:

   ```
   cat test.txt|dns2dic test.bin
   ```

   Reverse conversion using the dic2dns utility.

3. Place the resulting binary file in the directory where DPI will read it from:

   ```
   cp test.bin /var/lib/dpi/dns.bin
   ```

4. Create a service profile:

   ```
   fdpi_ctrl load profile --service 19 --profile.name test_193 --
   profile.json '{ "dns_list" : "/var/lib/dpi/dns.bin" }'
   ```

`max_profiles_serv19` — setting for the maximum number of profiles. Default is 32.

By default, the DNS response is sent to the interface from which the query originated (IN interface where the client query was received). Sending to the OUT interface is relevant for asymmetric DPI operation mode (only on outgoing traffic). Configured in `fastdpi.conf` with the `emit_direction=2` parameter

Management

Command format:

```
fdpi_ctrl [command] --service 19 [options list] [login or vchannel]
```

Enable service:

```
fdpi_ctrl load --service 19 --profile.name test_193 --login test
#or
```

```
fdpi_ctrl load --service 19 --profile.name test_193 --vchannel 1
```

Disable service:

```
fdpi_ctrl del --service 19 --profile.name test_193 --login test
#or
fdpi_ctrl del --service 19 --profile.name test_193 --vchannel 1
```

Searching for subscribers with the service assigned to the specified profile name:

```
fdpi_ctrl list all --service 19 --profile.name test_193
```

Deleting a named profile (there must be no subscribers using it):

```
fdpi_ctrl del profile --service 19  --profile.name test_193
```

Modifying service (profile) settings (new settings will be applied to all subscribers with the specified service profile):

```
fdpi_ctrl load profile --service 19 --profile.name test_193 --profile.json
'{ "dns_list" : "/var/lib/dpi/dns.bin" }'
```