RPZ block list Documentation

http://www.mypdns.org/

May 25, 2019

Contents

CHAPTER 1

Start

CHAPTER 2

Welcome to www.mypdns.org

You have probably been wondering why we are hiding who own's our domains used to run and maintain the RPZ zones.

The reason is actually rather simple, and yet very very sad, and the very reason why this project have been born.

The free and open DNS service is violating laws in many countries like ex. Denmark, which actually isn't a democratic country.... there are lot's of censuring going on and laws that says you have to block access to 10000's of domains if you have a open DNS server. Some of these domain are applied by law other by companies like Sony and interest organizations.

Another reason we are here is because it's bad for the free democracy that commercial companies and governments tracking all of you're activities on all kind of devices with internet access, YES even your TV and Radio collect data about you.

2.1 Our domains:

- matrix.rocks
- mypdns.cloud
- mypdns.com
- mypdns.org

A simple whois mypdns.cloud you'll get: .. code-block:

```
Domain Name: mypdns.cloud
Registry Domain ID: D92572D3CC69F46EC9EE510B585D4B0A5-NSR
Registrar WHOIS Server: whois.namesilo.com
```

(continues on next page)

(continued from previous page)

```
Registrar URL: www.namesilo.com
Updated Date: 2019-01-03T02:09:55Z
Creation Date: 2018-10-04T18:33:11Z
Registry Expiry Date: 2028-10-04T18:33:11Z
Registrar: NameSilo, LLC
Registrar IANA ID: 1479
Registrar Abuse Contact Email:
Registrar Abuse Contact Phone:
Domain Status: clientTransferProhibited https://icann.org/epp
 ↪#clientTransferProhibited
```

2.2 The WHOIS

Hopefully some usefull documentation for our RPZ zones and howto’s

2.3 Quiclist:

```
.rpz-passthru (Trigger PASSTHRU Action)
.rpz-drop (Drop Containing (IP))
google.com CNAME . (return NXDOMAIN for google.com)
*.google.com CNAME . (return NXDOMAIN for all subdomains in google.
 ↪com)
24.0.0.0.127.rpz-client-ip CNAME rpz-drop (Drop all requests from␣
 ↪clients in network 127.0.0.1/24)
.32.2.3.4.10.rpz-ip CNAME . (NXDOMAIN)
*.google.com.rpz-nsdname (NXDOMAIN all responses that comes from␣
 ↪dns server with name *.google.com)
```

CHAPTER 3

Support

The easiest way to get help with the project is to join the #mypdns channel on oftc. We hang out there and you can get real-time help with your projects.

The other good way is to open an issue on Github if you found a bug we couldn't fix on IRC

- oftc: irc://oftc.net
- Github: https://github.com/spirillen/rpz-block-list/issues

A lovely no -log in, -cookie option is out very own Trac where you can file a ticket or read the WIKI

CHAPTER 4

RPZ records

This document is written in corresponding to DNS Response Policy Zones (RPZ) and should meet the requerements from RFC 1034

Or as we like to think the RPZ should have standed for `Realtime Privacy Zone` :)

4.1 Quiclist:

```
 .rpz-passthru       (Trigger PASSTHRU Action)
 32.53.0.0.127.rpz-ip              CNAME rpz-drop. ;(Drop all␣
↪queries to IP 127.0.0.53)
 google.com                        CNAME .         ;(Return␣
↪NXDOMAIN for google.com)
 *.google.com                      CNAME .         ;(Return␣
↪NXDOMAIN for all subdomains in google.com)
 24.0.0.0.127.rpz-client-ip        CNAME rpz-drop. ;(Drop all␣
↪requests from clients in network 127.0.0.1/24)
 32.2.3.4.10.rpz-ip                CNAME .         ;(Return␣
↪NXDOMAIN response to all queries that would have lead to the IP␣
↪`10.4.3.2`)
 *.google.com.rpz-nsdname          CNAME .         ;(NXDOMAIN all␣
↪responses that comes from dns server within name *.google.com)
```

For IPv6 you replace `::` with `.zz.` and use `*.` as cname value

4.2 The “NXDOMAIN” Action (CNAME .)

A single resource record (RR) consisting of a CNAME whose target is the root domain (.) will cause a response of NXDOMAIN to be returned. This is the most commonly used RPZ action.

```
$ORIGIN RPZ.EXAMPLE.ORG.
example.com                       CNAME .        ; return NXDOMAIN
*.example.com                     CNAME .        ; return NXDOMAIN
```

4.3 The “NODATA” Action (CNAME *.)

```
$ORIGIN RPZ.EXAMPLE.ORG.
example.com                       CNAME *. ; return NODATA
*.example.com                     CNAME *. ; return NODATA
```

4.4 The “PASSTHRU” Action (CNAME rpz-passthru.)

It is sometimes necessary to exempt some DNS responses from a policy rule that covers an entire domain or a large IP address block. Exempting some clients of a DNS resolver from all RPZ rewriting can also be useful for research into attackers and for debugging. The PASSTHRU action is intended to override other, usually more general policies. The trigger for the PASSTHRU action MUST have a higher precedence than the policies that it should override (see Section 5, Precedence Rules).

In the example below, the first PASSTHRU record exempts requests for a particular host from the NXDOMAIN policy action of the subsequent records. The second PASSTHRU record exempts responses to the DNS client at 192.0.2.1 from being modified:

```
$ORIGIN RPZ.EXAMPLE.ORG.
ok.example.com                    CNAME rpz-passthru.
32.1.2.0.192.rpz-client-ip        CNAME rpz-passthru.
48.zz.101.db8.2001.rpz-client-ip  CNAME rpz-passthru.
example.com                       CNAME .
*.example.com                     CNAME .
```

4.5 The “DROP” Action (CNAME rpz-drop.)

The TCP-Only action is specified by a CNAME whose target is “rpz-tcp-only”. It changes UDP responses to short, truncated DNS responses that require the DNS client to try again with TCP. It is used to mitigate distributed DNS reflection attacks and is similar to the “slip” parameter of DNS Response Rate Limiting (RRL) [ISC-RRL].

```
$ORIGIN RPZ.EXAMPLE.ORG.
example.com             CNAME   rpz-tcp-only.
```

4.6 The “Local Data” Action (arbitrary RR types)

A set of RRsets with a common trigger owner name (see Section 4) that includes neither a special CNAME RPZ encoding of an action nor one of the problematic record types listed below specifies data to be used to generate synthetic DNS responses. The most common Local Data is a CNAME RR pointing to a walled garden, although other record types are also used.

```
$ORIGIN RPZ.EXAMPLE.ORG.
bad1.example.com                  CNAME garden.example.net.
bad2.example.com                  A     garden-web.example.net.
bad2.example.com                  MX    garden-mail.example.net.
32.3.2.0.192.rpz-client-ip        A     quarantine.example.net.
48.zz.101.db8.2001.rpz-client-ip  CNAME quarantine.example.org.
```

Note that because an RPZ is a valid DNS zone, if the action of a policy rule contains a CNAME RR, then no other RRs are allowed for that owner name (trigger).

The special RPZ encodings which are not to be taken as Local Data are CNAMEs with targets that are:

- “.” (NXDOMAIN action),
- “*.” (NODATA action),
- a top level domain starting with “rpz-“,
- a child of a top level domain starting with “rpz-“.

The problematic types and records which also do not encode Local Data actions include:

- SOA records,
- NS records,
- DNAME records,
- all DNSSEC-related records (see RFC 4034).

When a Local Data policy rule matches, the RRsets of Local Data are used to generate the response as if they comprised all of the authoritative data for the QNAME. If the requested type (QTYPE) is ANY, then all of these Local Data RRsets are returned. Otherwise, the RRset of the requested RR type is returned, or a CNAME is returned if it is available. If no CNAME nor RRset of the requested type is available, then the response is normally NODATA (ANCOUNT=0). Using the example above, if client 192.0.2.3 asks for MX records, it will receive NODATA, because the policy rule with the matching Client IP Address trigger contains RRsets of RRtype A but none of RRtype MX.

This normal NODATA response when there are no Local Data records of the requested type can be changed with the LOCAL-DATA-OR-PASSTHRU or LOCAL-DATA-OR-DISABLED overrides described in Section 6.1.

A special form of Local Data involves a CNAME RR with a wildcarded target name. Wildcards are not valid as CNAME targets in ordinary DNS zones. However, a wildcard in an RPZ Local Data CNAME target causes the matching QNAME to be prepended to the target in the rewritten response, which communicates this QNAME value to the walled garden DNS server for that DNS server's logs.

For example a policy Local Data action of "CNAME *.EXAMPLE.COM" applied to a QNAME of "EVIL.EXAMPLE.ORG." will result in a synthetic response that starts with the RR "EVIL.EXAMPLE.ORG CNAME EVIL.EXAMPLE.ORG.EXAMPLE.COM". Resolving the CNAME target "EVIL.EXAMPLE.ORG.EXAMPLE.COM" into an RRset of the originally requested type generally requires sending a request for that type and a QNAME of "EVIL.EXAMPLE.ORG.EXAMPLE.COM" to a DNS server for the walled garden, "EXAMPLE.COM". As usual when a CNAME is encountered while computing a response, the response from the walled garden DNS server concerning "EVIL.EXAMPLE.ORG.EXAMPLE.COM" determines the rest of the final rewritten response.

In the example below, a client that asks for A RRs for "BAD.EXAMPLE.COM" will receive a response starting with "BAD.EXAMPLE.COM CNAME BAD.EXAMPLE.COM.GARDEN.EXAMPLE.NET". The DNS server using RPZ will then probably try to resolve "BAD.EXAMPLE.COM.GARDEN.EXAMPLE.NET" by requesting A RRs from the authority for "GARDEN.EXAMPLE.NET". That authority should answer with NODATA, NXDOMAIN, or an A RRset, but in any case can log the request to show that a request for "BAD.EXAMPLE.COM" has been received.

```
$ORIGIN RPZ.EXAMPLE.ORG.
bad.example.com                   CNAME *.garden.example.net.
```

4.7 The "Client IP Address" Trigger (.rpz-client-ip)

The IP addresses of DNS clients sending requests can be used as triggers, which can be useful for disabling RPZ rewriting for DNS clients used for testing or investigating, or for quarantining compromised clients. Client IP Address policy RRsets have owner names that are subdomains of "rpz-client-ip" relativized to the RPZ apex name, preceded by an encoded IP address or block of addresses.

For example, the following would drop all requests from clients in 192.0.2.0/24 and give truthful answers to requests from a client at 2001:db8::3.

```
$ORIGIN RPZ.EXAMPLE.ORG.
24.0.2.0.192.rpz-client-ip        CNAME rpz-drop.
128.3.zz.db8.2001.rpz-client-ip   CNAME rpz-passthru.
```

4.8 The “QNAME” Trigger (“example.com”)

The QNAME policy trigger matches requested domains, that is, the QNAME field of the question sections in DNS requests and responses. (See [RFC1035].) The owner name of an RPZ QNAME policy RRset is the relativized name of the domain name about which policy is being expressed. For example, if the RPZ apex name is “RPZ.EXAMPLE.ORG”, an RRset at “EXAMPLE.COM.RPZ.EXAMPLE.ORG” would affect responses to requests about “EXAMPLE.COM”.

Wildcards work as expected, so the owner name “*.EXAMPLE.COM.RPZ.EXAMPLE.ORG” would match queries for any subdomain of “EXAMPLE.COM”. To control the policy for both a name and its subdomains, two policy RRsets must be used, one for the domain itself and another for a wildcard subdomain. In the following example, queries for both “EXAMPLE.COM” and all subdomains of “EXAMPLE.COM” will result in synthetic NXDOMAIN responses.

```
$ORIGIN RPZ.EXAMPLE.ORG.
example.com                       CNAME .        ; return NXDOMAIN
*.example.com                     CNAME .        ; return NXDOMAIN
```

4.9 The “Response IP Address” Trigger (.rpz-ip)

The Response IP Address trigger matches IP addresses that would appear in the unaltered DNS response contents, specifically the RDATA of A or AAAA records in the answer sections of DNS responses. IP addresses in the authority and additional sections are not considered. Response IP Address policy RRsets have owner names that are subdomains of “rpz-ip” relativized to the RPZ apex name, and an encoded IP address or block of addresses. The IP address encodings are identical to those described in Section 4.1.1 for Client IP Address triggers.

For example, to force an NXDOMAIN response whenever a truthful response would contain an answer section A RRset having an address in 192.0.2.0/24 unless address 192.0.2.2 is present, the RPZ would contain these records:

```
$ORIGIN RPZ.EXAMPLE.ORG.
24.0.2.0.192.rpz-ip               CNAME .
32.2.2.0.192.rpz-ip               CNAME rpz-passthru.
48.zz.101.db8.2001.rpz-client-ip  CNAME rpz-passthru.
```

4.10 The “NSDNAME” Trigger (.rpz-nsdname)

The NSDNAME policy trigger matches name server names (NS RR) of all name servers in the data paths for all RRsets that would be present in the answer section of the unaltered DNS response.

The data path for a given answer RRset consists of all delegation points from (and including) the root zone down to the closest enclosing NS RRset for the owner name of that RRset. Names in the RDATA of answer RRs including CNAME, DNAME, SRV, and MX are not themselves directly relevant, but CNAME and DNAME target names are indirectly relevant if they cause RRsets to be added to the answer section, in which case it is the data paths of the added RRsets that matter. In the case of a DNAME answer, the owner name of an added synthetic CNAME is likely to differ from the target name in the DNAME RR. Recall also that the target of a CNAME is not added to the response if the QTYPE is ANY or CNAME or if this target cannot be resolved (e.g. NXDOMAIN or SERVFAIL errors).

NSDNAME policies are encoded as RRsets in subdomains of "rpz-nsdname" but otherwise are much like QNAME policies (Section 4.2). For example, to force an NXDOMAIN answer whenever a name server for the requested domain or one of its parents is "NS.EXAMPLE.COM", the RPZ would contain the following:

```
$ORIGIN RPZ.EXAMPLE.ORG.
ns.example.com.rpz-nsdname        CNAME .
```

The NS records used for this calculation are either delegations (NS RRs in the authority sections of answers from authorities for the parent zone) or authoritative data from the zone itself. An implementation MAY use either, both, or whichever is currently available. See Section 9.4 about some implementation considerations for this choice, and Section 12.5 about security considerations.

An RPZ implementation MAY be configurable to avoid checking all the way up to the root and to perform only partial NSDNAME checks; see Section 9.3 on "min-ns-dots".

4.11 The "NSIP" Trigger (.rpz-nsip)

The NSIP policy trigger matches name server addresses, that is A or AAAA RRs referenced by an NS RRset. NSIP is like NSDNAME (Section 4.4) except that the matching is by name server address rather than name server name. NSIP policies are expressed as subdomains of "rpz-nsip" and have the same subdomain naming convention as that described for encoding IP addresses in Response IP Address triggers (Section 4.1.1).

In a process similar to that for an NSDNAME trigger, an NSIP trigger is checked by considering all of the IP addresses for all of the name servers in the data paths for all RRsets that would be present in the answer section of the unaltered DNS response.

As with NSDNAME triggers, the data path for a given RRset consists of all delegation points from (and including) the root zone down to the closest enclosing NS RRset for the owner name of that RRset. Also like NSDNAME triggers, the RDATA in these RRsets (other than the IP addresses of the name servers) are not directly relevant.

For example, to force an NXDOMAIN answer whenever one of the name servers for the requested domain (QNAME) or one of its ancestors has an address in the 192.0.2.0/24 block, the RPZ would contain the following:

```
$ORIGIN RPZ.EXAMPLE.ORG.
24.0.2.0.192.rpz-nsip             CNAME .
48.zz.101.db8.2001.rpz-client-ip  CNAME rpz-tcp-only.
```

The NS, A, and AAAA records used for this calculation are either delegations and glue (RRs in authority and additional sections of answers from authorities for the parent zone) or authoritative data from the zone itself. As with NSIP, an implementation MAY use either, both, or whichever is currently available; see Section 9.4 on “ns-wait-recurse”.

An RPZ implementation MAY also be configurable to avoid checking all the way up to the root and to perform only partial NSIP checks; see Section 9.3 on “min-ns-dots”.