DNS Service Discovery

basic data infrastructure

Topsector Logistiek

Colophon

DNS Service Discovery

Author
Remco van 't Veer

January 2024

Table of contents

1 Service Discovery

In a situation where different parties want to share data with each other, it is important that they can find each other's data easily. This requires agreements on what will be shared and how, and where that data will come from.
In this memo, we focus on Service Discovery - a mechanism with which systems can find each other easily - and specifically discovery using DNS[1].

1 *Domain Name System, the system that links names to computer addresses*

2 DNS

DNS is one of the oldest standards on the internet that is still in use. Every organisation already uses DNS to make parts of their infrastructure public. Examples include email and websites. Further, DNS is the backbone of many federated services such as VoIP[2], LDAP[3] and XMPP[4] (note that email is also a federated service).

A DNS zone[5] is managed by an organisation itself, or the management is outsourced. This makes it possible to fill in "their part of the internet" freely, which includes "names". In these zones, further "sub-zones" can be created, which can be managed by their own departments, creating a hierarchical structure.

2 Voice over IP, telephony over the internet
3 Lightweight Directory Access Protocol, finding users' information
4 Extensible Messaging and Presence Protocol, widely used for chat and IoT applications
5 A specific part of the names in the DNS

3 How?

With email, it works - simplified - as follows. For every domain where people want to receive email, the organisation creates *MX* records[6] that refer to servers where email can be received. When sending an email, a DNS server is asked what *MX* records exist for the receiving domain and contact is made with that server to deliver the message. In addition, *TXT* records[7] may also have been created that contain cryptographic keys that can be used to prove the origin of the email.

The DNS record types are established. With this, *MX* records are intended for email and not for anything else. Fortunately, a *TXT* is free to use and multiple records can be registered with the same name, which enables lists of information to be retrieved via DNS. With email, for example, this is used if the first server to which an MX record refers is not reachable, the next one is tried.

We describe the different components for service discovery below. Then, in section 3.5 , we describe a concrete example of an organisation with multiple services.

The following ingredients are needed for service discovery:

3.1 Well-known subdomain

By introducing a "well-known subdomain", we can specify a point with a predictable name where it is possible to query what services are available; in other words, Service Discovery.

For example: `_bdi.acme-corp.com.` This can immediately be the main level of a DNS zone and assigned to the appropriate administrators.

Note that this is not a *hostname* but a *domainname*. A hostname, as used in websites and email, may not contain underscore (_) characters. The use of an underscore indicates that it is a "special" record.

3.2 Discovery

Now that we have a clear starting point, it can be set up so there is something to discover. We do this with *TXT* records because we have complete freedom there and can create even more "special" records in our "well-known subdomain". To do this, we followed the forms:

- `_bdi.<subdomain>` ***`TXT records`***
  Provides list of all types of services for this subdomain in the form below.

- `_<service>._<proto>._bdi.<subdomain>` ***`TXT records`***
  Where *proto* is the internet protocol used by the service (`tcp` or `udp`) and *service* is the service type (`ldap, mqtt` etc.). Provides list of all services of that type (instances) in the form below.

- `<instance>._<service>._<proto>._bdi.<subdomain>` ***`TXT records`***
  Where *instance* is a freely chosen name.

6 *MX stands for "mail exchanger"*
7 *"Free" text can be stored in TXT records*

3.3 SRV records

Now we know the actual services but do not yet know exactly where to find them. After all, underscores should not appear in hostnames and it is not desirable to create a direct *A* record[8] or *CNAME* record[9] for this server. That is why we switch to *SRV* records[10]. These records contain:

- *target*
  The hostname or IP address where this service can be reached.

- *port*
  The port number where this service can be reached.

- ***priority*** en ***weight***
  Values used to choose a record when multiple endpoints are available for this service.

If multiple *SRV* records records exist for a service, the endpoints referenced must provide exactly the same service. The *priority* and *weight* gegevens data are used to choose the best candidate from the list of records (see RFC 2782 for more information).

Now we know where to get information from, with which protocol, on which server, and via which port number.

.

3.4 TXT records

For some services, the data from the *SRV* record is not enough. For that, it is possible to create additional *TXT* records with the same name as the *SRV* record. What this contains depends on the specific service and the protocol used, but this could be things like the rights needed to use the service.

Information is indicated in these records as attributes with a name and a value that are linked with an `=` teken sign and attributes are separated by a `;` sign. Although all characters are possible in *TXT* records, restrictions on usage are often imposed by DNS service providers. Unfortunately, these rules are not the same for all providers, but all providers will support all numbers, letters, `+`, `/`, `=`, `;` characters, and spaces[11].

An example: `quality=high; resolution=seconds`

8 Address record, a hostname to an IP address
9 Common name record, a hostname alias that in turn refers to another hostname
10 Service records describe where a server is
11 Letters and numbers mean the ASCII standard letters and numbers

3.5 An example

We take the ACME Corporation as an example. As their "well-known subdomain", they have:

- `_bdi.acme-corp.com`

The corresponding *TXT* records tell that they have SPARQL and MQTT end points.

- `_bdi.acme-corp.com. 3600 IN TXT`[12]
  `"_sparql._tcp._bdi.acme-corp.com"`

- `_bdi.acme-corp.com. 3600 IN TXT`
  `"_mqtt._tcp._bdi.acme-corp.com"`

We zoom in on MQTT, where we find:

- `_mqtt._tcp._bdi.acme-corp.com. 3600 IN TXT`
  `"warehouse-status-events._mqtt._tcp._bdi.acme-corp.com"`

- `_mqtt._tcp._bdi.acme-corp.com. 3600 IN TXT`
  `"logistic-events._mqtt._tcp._bdi.acme-corp.com"`

We find server with the *SRV* record:

- `logistic-events._mqtt._tcp._bdi.acme-corp.com. 3600 IN SRV`
  `"1 1 443 mqqt.logistics.acme-corp.com"`[13]

- `logistic-events._mqtt._tcp._bdi.acme-corp.com. 3600 IN SRV`
  `"2 1 443 mqqt-backup.logistics.acme-corp.com"`

And extra information:

- `logistic-events._mqtt._tcp._bdi.acme-corp.com. 3600 IN TXT`
  `"queue=main; auth=ishare"`

We now know that for *MQTT* messages we need to contact *mqqt.logistics.acme-corp.com com (and if this is not available mqqt-backup.logistics.acme-corp.com*) on port 443, that everything of interest can be found on queue main and that authorisation is arranged via *iSHARE*.

3.6 Origin

This is loosely based on the DNS-SD[14]. However, this standard cannot be used because it uses *PTR* records which are very impractical to use because they are mainly used to stop unwanted email and therefore are offered by few DNS service providers. Furthermore, it assumes more control over the record than providers allow; possible characters in name and value, for example.

12 The number 3600 indicates that the record may be retained for one hour (3600 seconds)
13 The numbers 1, 1 and 433 are the priority, weight and port of this service respectively
14 See also RFC 6763

4 Security

The DNS platform is secured by DNSSEC[15]. The use of DNSSEC is required for secure implementation of the mechanism described here. DNSSEC is a widely supported standard and all DNS service providers support it.

15 *Domain Name System Security Extensions, this prevents someone else from posing as the organisation affected and thereby providing false information or stealing login data.*

5 In conclusion

The mechanism described here is a starting point for a service discovery system and is therefore not complete. In particular, section 3.4 on the use of *TXT* records records can be elaborated further by *service type* and there is probably a need for agreements for the *instance* names in section 3.2.

DNS is a key component for the proper functioning of the internet and is battle-hardened. Numerous federated services have been using DNS as a registry successfully for decades and use it to implement forms of service discovery. Management is easily distributed hierarchically by means of zones. These characteristics make DNS a prime candidate for implementing service discovery.

Topsector Logistiek
Ezelsveldlaan 59
2611 RV Delft
+31 15 251 65 65
www.topsectorlogistiek.nl

Topsector Logistiek