Use cases

Network-provided proxy discovery
Related proxy discovery
Proxy applicability discovery (split DNS, etc)
Network-provided proxy discovery

Network (ISP, carrier) provides a proxy that can assist in mobility (AT-SSS) or privacy (act as a first-hop Private Relay proxy)
Related proxy discovery

*Client knows about CONNECT support, but proxy also supports CONNECT-UDP and CONNECT-IP*

I also have URLs for UDP and IP proxying
Proxy applicability discovery (split DNS, etc)

A proxy only allows access to certain hosts, or from certain users (not an "open" proxy)

I only allow access to corp.example.com
Existing alternatives

WPAD and PAC fill similar roles

Not actually IETF standards

Requires running javascript parsing, with many possible security pitfalls

Would need to be significantly extended to know about new proxy types that use URLs

Let's not do this, please!
So what is a Provisioning Domain, and how can it help?
PvDs

Defined by MIF and INTAREA

RFC 7556 defines Provisioning Domains (PvDs) as "consistent sets of network configuration information"

RFC 8801 defines PvD discovery & additional data

- IPv6 RA advertisement
- JSON info fetched over HTTP
But what is a PvD?

Your home Wi-Fi uplink over a particular ISP

A carrier network configuration

A VPN configuration

...

A proxy configuration too!
PvDs for proxies

This document discusses two mechanisms

1. Enumerating proxy URLs related to a PvD

2. Fetching a PvD configuration from an HTTP-based proxy
Proxy list in PvD JSON

The "proxies" key is an array of URLs

```json
{
  "identifier": "company.example.org."
  "expires": "2023-06-23T06:00:00Z",
  "prefixes": [
    "2001:db8:cafe::/48"
  ],
  "proxies": [
    "https://proxy.example.org",
    "https://proxy.example.org/masque{?target_host,target_port}"
  ]
}
```

Type of proxy here is implicit in the URL (scheme and URI template variables imply SOCKS vs HTTP CONNECT vs CONNECT-UDP)

❓ Should we define more formal types, as a more complex dictionary format?
Fetching the PvD for a proxy

Since each proxy has its own conceptual PvD, it can use the PvD additional information to indicate related DNS zones, or even other proxies.

With a proxy URL of https://proxy.example.org/masque{?target_host,target_port}:

:method = GET
:scheme = https
:authority = proxy.example.org
:path = /masque
:accept = application/pvd+json

With a proxy URL of https://proxy.example.org:

:method = GET
:scheme = https
:authority = proxy.example.org
:path = /
:accept = application/pvd+json
Split DNS zones

Existing DNS zones key can indicate that this proxy only serves specific domains, providing a remote-access VPN configuration

:status = 200
content-type = application/pvd+json
content-length = 135

{
    "identifier": "proxy.example.org.",
    "expires": "2023-06-23T06:00:00Z",
    "prefixes": [],
    "dnsZones": ["internal.example.org"]
}

❓ What other attributes would make sense to add here?
Next steps

Adopt this in INTAREA?

Would it fit better elsewhere (HTTP, etc)? INTAREA is where PvDs was developed, and also was the home for previous SOCKS discussions.

Interop testing!