# » powerdns_record

Provides a PowerDNS record resource.

## » Example Usage

Note that PowerDNS may internally lowercase certain records (e.g. CNAME and AAAA), which may lead to resources being marked for a change in every single plan/apply.

### » A record example

For the v1 API (PowerDNS version 4):

```
# Add a record to the zone
resource "powerdns_record" "foobar" {
  zone    = "example.com."
  name    = "www.example.com."
  type    = "A"
  ttl     = 300
  records = ["192.168.0.11"]
}
```

### » PTR record example

An example creating PTR record:

```
# Add PTR record to the zone
resource "powerdns_record" "foobar" {
  zone    = "0.10.in-addr.arpa."
  name    = "10.0.0.10.in-addr.arpa."
  type    = "PTR"
  ttl     = 300
  records = ["www.example.com."]
}
```

### » Automatically set PTR record for A/AAAA records

**Deprecation warning:** *set_ptr* feature is set to be deprecated in PowerDNS v4.3.0

PowerDNS API v4.2.0 offers a feature to automatically create corresponding PTR record for the A/AAAA record. Existing PTR records with the same

name are replaced. If no matching reverse zone is found, resource creation will fail. You can use `powerdns_zone` resource to create the reverse zone.

**Warning:** Using *set_ptr:true* will not automatically remove the PTR record when A/AAAA record is deleted. You should create PTR zone using `powerdns_zone` and manage PTR records using `powerdns_record`, rather than using *set_ptr*. With upcoming *set_ptr* deprecation, this will be the only way of maintaining PTR records **via this provider**.

Here is an example of creating A record along with corresponding PTR record:

```
resource "powerdns_record" "foobar" {
  zone    = "example.com."
  name    = "www.example.com"
  type    = "A"
  ttl     = 300
  set_ptr = true
  records = ["192.168.0.11"]
}
```

For the legacy API (PowerDNS version 3.4):

```
# Add a record to the zone
resource "powerdns_record" "foobar" {
  zone    = "example.com."
  name    = "www.example.com."
  type    = "A"
  ttl     = 300
  records = ["192.168.0.11"]
}
```

## » Argument Reference

The following arguments are supported:

- `zone` - (Required) The name of zone to contain this record.
- `name` - (Required) The name of the record.
- `type` - (Required) The record type.
- `ttl` - (Required) The TTL of the record.
- `records` - (Required) A string list of records.
- `set_ptr` - (Optional) [***Deprecated in PowerDNS 4.3.0***] A boolean (true/false), determining whether API server should automatically create PTR record in the matching reverse zone. Existing PTR records are replaced. If no matching reverse zone, an error is thrown.

### » Attribute Reference

The id of the resource is a composite of the record name and record type, joined by a separator - `:::`.

For example, record `foo.test.com.` of type `A` will be represented with the following `id`: `foo.test.com.:::A`

### » Importing

An existing record can be imported into this resource by supplying both the record id and zone name it belongs to. If the record or zone is not found, or if the record is of a different type or in a different zone, an error will be returned.

For example:

```
$ terraform import powerdns_record.test-a '{"zone": "test.com.", "id": "foo.test.com.:::A"}'
```

For more information on how to use terraform's `import` command, please refer to terraform's core documentation.

## » powerdns_zone

Provides a PowerDNS zone.

### » Example Usage

For the v1 API (PowerDNS version 4):

```
# Add a zone
resource "powerdns_zone" "foobar" {
  name    = "example.com."
  kind    = "Native"
  nameservers = ["ns1.example.com.", "ns2.example.com."]
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of zone.
- `kind` - (Required) The kind of the zone.
- `nameservers` - (Required) The zone nameservers.

- `soa_edit_api` - (Optional) This should map to one of the supported API values *or* in case you wish to remove the setting, set this argument as `\"\"` (that will translate to the API value `""`).

## » Importing

An existing zone can be imported into this resource by supplying the zone name. If the zone is not found, an error will be returned.

For example, to import zone `test.com.`:

```
$ terraform import powerdns_zone.test test.com.
```

For more information on how to use terraform's `import` command, please refer to terraform's core documentation.