» fastly_ip_ranges

Use this data source to get the IP ranges of Fastly edge nodes.

» Example Usage

```
data "fastly_ip_ranges" "fastly" {}

resource "aws_security_group" "from_fastly" {
  name = "from_fastly"

  ingress {
    from_port   = "443"
    to_port     = "443"
    protocol    = "tcp"
    cidr_blocks = ["${data.fastly_ip_ranges.fastly.cidr_blocks}"]
  }
}
```

» Attributes Reference

- `cidr_blocks` - The lexically ordered list of CIDR blocks.

» fastly_service_v1

Provides a Fastly Service, representing the configuration for a website, app, API, or anything else to be served through Fastly. A Service encompasses Domains and Backends.

The Service resource requires a domain name that is correctly set up to direct traffic to the Fastly service. See Fastly's guide on Adding CNAME Records on their documentation site for guidance.

» Example Usage

Basic usage:

```
resource "fastly_service_v1" "demo" {
  name = "demofastly"

  domain {
    name    = "demo.notexample.com"
```

```
    comment = "demo"
  }

  backend {
    address = "127.0.0.1"
    name    = "localhost"
    port    = 80
  }

  force_destroy = true
}
```

Basic usage with an Amazon S3 Website and that removes the `x-amz-request-id` header:

```
resource "fastly_service_v1" "demo" {
  name = "demofastly"

  domain {
    name    = "demo.notexample.com"
    comment = "demo"
  }

  backend {
    address = "demo.notexample.com.s3-website-us-west-2.amazonaws.com"
    name    = "AWS S3 hosting"
    port    = 80
  }

  header {
    destination = "http.x-amz-request-id"
    type        = "cache"
    action      = "delete"
    name        = "remove x-amz-request-id"
  }

  gzip {
    name          = "file extensions and content types"
    extensions    = ["css", "js"]
    content_types = ["text/html", "text/css"]
  }

  default_host = "${aws_s3_bucket.website.name}.s3-website-us-west-2.amazonaws.com"

  force_destroy = true
}
```

```
resource "aws_s3_bucket" "website" {
  bucket = "demo.notexample.com"
  acl    = "public-read"

  website {
    index_document = "index.html"
    error_document = "error.html"
  }
}
```

Basic usage with custom VCL (must be enabled on your Fastly account):

```
resource "fastly_service_v1" "demo" {
  name = "demofastly"

  domain {
    name    = "demo.notexample.com"
    comment = "demo"
  }

  backend {
    address = "127.0.0.1"
    name    = "localhost"
    port    = 80
  }

  force_destroy = true

  vcl {
    name    = "my_custom_main_vcl"
    content = "${file("${path.module}/my_custom_main.vcl")}"
    main    = true
  }

  vcl {
    name    = "my_custom_library_vcl"
    content = "${file("${path.module}/my_custom_library.vcl")}"
  }
}
```

Note: For an AWS S3 Bucket, the Backend address is `<domain>.s3-website-<region>.amazonaws.com`. The `default_host` attribute should be set to `<bucket_name>.s3-website-<region>.amazonaws.com`. See the Fastly documentation on Amazon S3.

» Argument Reference

The following arguments are supported:

- `name` - (Required) The unique name for the Service to create.
- `domain` - (Required) A set of Domain names to serve as entry points for your Service. Defined below.
- `backend` - (Optional) A set of Backends to service requests from your Domains. Defined below. Backends must be defined in this argument, or defined in the `vcl` argument below
- `condition` - (Optional) A set of conditions to add logic to any basic configuration object in this service. Defined below.
- `cache_setting` - (Optional) A set of Cache Settings, allowing you to override when an item is not to be cached based on an above `condition`. Defined below
- `gzip` - (Required) A set of gzip rules to control automatic gzipping of content. Defined below.
- `header` - (Optional) A set of Headers to manipulate for each request. Defined below.
- `healthcheck` - (Optional) Automated healthchecks on the cache that can change how fastly interacts with the cache based on its health.
- `default_host` - (Optional) The default hostname.
- `default_ttl` - (Optional) The default Time-to-live (TTL) for requests.
- `force_destroy` - (Optional) Services that are active cannot be destroyed. In order to destroy the Service, set `force_destroy` to `true`. Default `false`.
- `request_setting` - (Optional) A set of Request modifiers. Defined below
- `s3logging` - (Optional) A set of S3 Buckets to send streaming logs too. Defined below.
- `papertrail` - (Optional) A Papertrail endpoint to send streaming logs too. Defined below.
- `sumologic` - (Optional) A Sumologic endpoint to send streaming logs too. Defined below.
- `gcslogging` - (Optional) A gcs endpoint to send streaming logs too. Defined below.
- `syslog` - (Optional) A syslog endpoint to send streaming logs too. Defined below.
- `logentries` - (Optional) A logentries endpoint to send streaming logs too. Defined below.
- `response_object` - (Optional) Allows you to create synthetic responses that exist entirely on the varnish machine. Useful for creating error or maintenance pages that exists outside the scope of your datacenter. Best when used with Condition objects.
- `vcl` - (Optional) A set of custom VCL configuration blocks. The ability to upload custom VCL code is not enabled by default for new Fastly accounts (see the Fastly documentation for details).

The `domain` block supports:

- `name` - (Required) The domain to which this Service will respond.
- `comment` - (Optional) An optional comment about the Domain.

The `backend` block supports:

- `name` - (Required, string) Name for this Backend. Must be unique to this Service.
- `address` - (Required, string) An IPv4, hostname, or IPv6 address for the Backend.
- `auto_loadbalance` - (Optional, boolean) Denotes if this Backend should be included in the pool of backends that requests are load balanced against. Default `true`.
- `between_bytes_timeout` - (Optional) How long to wait between bytes in milliseconds. Default `10000`.
- `connect_timeout` - (Optional) How long to wait for a timeout in milliseconds. Default `1000`
- `error_threshold` - (Optional) Number of errors to allow before the Backend is marked as down. Default `0`.
- `first_byte_timeout` - (Optional) How long to wait for the first bytes in milliseconds. Default `15000`.
- `max_conn` - (Optional) Maximum number of connections for this Backend. Default `200`.
- `port` - (Optional) The port number on which the Backend responds. Default `80`.
- `request_condition` - (Optional, string) Name of already defined `condition`, which if met, will select this backend during a request.
- `use_ssl` - (Optional) Whether or not to use SSL to reach the backend. Default `false`.
- `max_tls_version` - (Optional) Maximum allowed TLS version on SSL connections to this backend.
- `min_tls_version` - (Optional) Minimum allowed TLS version on SSL connections to this backend.
- `ssl_ciphers` - (Optional) Comma separated list of OpenSSL Ciphers to try when negotiating to the backend.
- `ssl_ca_cert` - (Optional) CA certificate attached to origin.
- `ssl_client_cert` - (Optional) Client certificate attached to origin. Used when connecting to the backend.
- `ssl_client_key` - (Optional) Client key attached to origin. Used when connecting to the backend.
- `ssl_check_cert` - (Optional) Be strict about checking SSL certs. Default `true`.
- `ssl_hostname` - (Optional, deprecated by Fastly) Used for both SNI during the TLS handshake and to validate the cert.
- `ssl_cert_hostname` - (Optional) Overrides ssl_hostname, but only for cert verification. Does not affect SNI at all.

- `ssl_sni_hostname` - (Optional) Overrides ssl_hostname, but only for SNI in the handshake. Does not affect cert validation at all.
- `shield` - (Optional) The POP of the shield designated to reduce inbound load.
- `weight` - (Optional) The portion of traffic to send to this Backend. Each Backend receives `weight / total` of the traffic. Default `100`.
- `healthcheck` - (Optional) Name of a defined `healthcheck` to assign to this backend.

The `condition` block supports allows you to add logic to any basic configuration object in a service. See Fastly's documentation "About Conditions" for more detailed information on using Conditions. The Condition `name` can be used in the `request_condition`, `response_condition`, or `cache_condition` attributes of other block settings.

- `name` - (Required) The unique name for the condition.
- `statement` - (Required) The statement used to determine if the condition is met.
- `type` - (Required) Type of condition, either `REQUEST` (req), `RESPONSE` (req, resp), or `CACHE` (req, beresp).
- `priority` - (Optional) A number used to determine the order in which multiple conditions execute. Lower numbers execute first. Default `10`.

The `cache_setting` block supports:

- `name` - (Required) Unique name for this Cache Setting.
- `action` - (Optional) One of `cache`, `pass`, or `restart`, as defined on Fastly's documentation under "Caching action descriptions".
- `cache_condition` - (Optional) Name of already defined `condition` used to test whether this settings object should be used. This `condition` must be of type `CACHE`.
- `stale_ttl` - (Optional) Max "Time To Live" for stale (unreachable) objects.
- `ttl` - (Optional) The Time-To-Live (TTL) for the object.

The `gzip` block supports:

- `name` - (Required) A unique name.
- `content_types` - (Optional) The content-type for each type of content you wish to have dynamically gzip'ed. Example: `["text/html", "text/css"]`.
- `extensions` - (Optional) File extensions for each file type to dynamically gzip. Example: `["css", "js"]`.
- `cache_condition` - (Optional) Name of already defined `condition` controlling when this gzip configuration applies. This `condition` must be of type `CACHE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `Header` block supports adding, removing, or modifying Request and Re-

sponse headers. See Fastly's documentation on Adding or modifying headers on HTTP requests and responses for more detailed information on any of the properties below.

- `name` - (Required) Unique name for this header attribute.
- `action` - (Required) The Header manipulation action to take; must be one of `set`, `append`, `delete`, `regex`, or `regex_repeat`.
- `type` - (Required) The Request type on which to apply the selected Action; must be one of `request`, `fetch`, `cache` or `response`.
- `destination` - (Required) The name of the header that is going to be affected by the Action.
- `ignore_if_set` - (Optional) Do not add the header if it is already present. (Only applies to the `set` action.). Default `false`.
- `source` - (Optional) Variable to be used as a source for the header content. (Does not apply to the `delete` action.)
- `regex` - (Optional) Regular expression to use (Only applies to the `regex` and `regex_repeat` actions.)
- `substitution` - (Optional) Value to substitute in place of regular expression. (Only applies to the `regex` and `regex_repeat` actions.)
- `priority` - (Optional) Lower priorities execute first. Default: `100`.
- `request_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `REQUEST`.
- `cache_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `CACHE`.
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `healthcheck` block supports:

- `name` - (Required) A unique name to identify this Healthcheck.
- `host` - (Required) The Host header to send for this Healthcheck.
- `path` - (Required) The path to check.
- `check_interval` - (Optional) How often to run the Healthcheck in milliseconds. Default `5000`.
- `expected_response` - (Optional) The status code expected from the host. Default `200`.
- `http_version` - (Optional) Whether to use version 1.0 or 1.1 HTTP. Default `1.1`.
- `initial` - (Optional) When loading a config, the initial number of probes to be seen as OK. Default `2`.
- `method` - (Optional) Which HTTP method to use. Default `HEAD`.
- `threshold` - (Optional) How many Healthchecks must succeed to be considered healthy. Default `3`.
- `timeout` - (Optional) Timeout in milliseconds. Default `500`.
- `window` - (Optional) The number of most recent Healthcheck queries to keep for this Healthcheck. Default `5`.

The `request_setting` block allow you to customize Fastly's request handling, by defining behavior that should change based on a predefined `condition`:

- `name` - (Required) The domain for this request setting.
- `request_condition` - (Optional) Name of already defined `condition` to determine if this request setting should be applied.
- `max_stale_age` - (Optional) How old an object is allowed to be to serve `stale-if-error` or `stale-while-revalidate`, in seconds.
- `force_miss` - (Optional) Force a cache miss for the request. If specified, can be `true` or `false`.
- `force_ssl` - (Optional) Forces the request to use SSL (Redirects a non-SSL request to SSL).
- `action` - (Optional) Allows you to terminate request handling and immediately perform an action. When set it can be `lookup` or `pass` (Ignore the cache completely).
- `bypass_busy_wait` - (Optional) Disable collapsed forwarding, so you don't wait for other objects to origin.
- `hash_keys` - (Optional) Comma separated list of varnish request object fields that should be in the hash key.
- `xff` - (Optional) X-Forwarded-For, should be `clear`, `leave`, `append`, `append_all`, or `overwrite`. Default `append`.
- `timer_support` - (Optional) Injects the X-Timer info into the request for viewing origin fetch durations.
- `geo_headers` - (Optional) Injects Fastly-Geo-Country, Fastly-Geo-City, and Fastly-Geo-Region into the request headers.
- `default_host` - (Optional) Sets the host header.

The `s3logging` block supports:

- `name` - (Required) A unique name to identify this S3 Logging Bucket.
- `bucket_name` - (Optional) An optional comment about the Domain.
- `s3_access_key` - (Required) AWS Access Key of an account with the required permissions to post logs. It is **strongly** recommended you create a separate IAM user with permissions to only operate on this Bucket. This key will be not be encrypted. You can provide this key via an environment variable, `FASTLY_S3_ACCESS_KEY`.
- `s3_secret_key` - (Required) AWS Secret Key of an account with the required permissions to post logs. It is **strongly** recommended you create a separate IAM user with permissions to only operate on this Bucket. This secret will be not be encrypted. You can provide this secret via an environment variable, `FASTLY_S3_SECRET_KEY`.
- `path` - (Optional) Path to store the files. Must end with a trailing slash. If this field is left empty, the files will be saved in the bucket's root path.
- `domain` - (Optional) If you created the S3 bucket outside of `us-east-1`, then specify the corresponding bucket endpoint. Example: `s3-us-west-2.amazonaws.com`.
- `period` - (Optional) How frequently the logs should be transferred, in seconds. Default `3600`.

- `gzip_level` - (Optional) Level of GZIP compression, from `0-9`. `0` is no compression. `1` is fastest and least compressed, `9` is slowest and most compressed. Default `0`.
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (`%h %l %u %t %r %>s`)
- `message_type` - (Optional) How the message should be formatted; one of: `classic`, `loggly`, `logplex` or `blank`. Default `classic`.
- `timestamp_format` - (Optional) `strftime` specified timestamp formatting (default `%Y-%m-%dT%H:%M:%S.000`).
- `redundancy` - (Optional) The S3 redundancy level. Should be formatted; one of: `standard`, `reduced_redundancy` or null. Default `null`.
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `papertrail` block supports:

- `name` - (Required) A unique name to identify this Papertrail endpoint.
- `address` - (Required) The address of the Papertrail endpoint.
- `port` - (Required) The port associated with the address where the Papertrail endpoint can be accessed.
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (`%h %l %u %t %r %>s`)
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `sumologic` block supports:

- `name` - (Required) A unique name to identify this Sumologic endpoint.
- `url` - (Required) The URL to Sumologic collector endpoint
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (`%h %l %u %t %r %>s`)
- `format_version` - (Optional) The version of the custom logging format used for the configured endpoint. Can be either 1 (the default, version 1 log format) or 2 (the version 2 log format).
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.
- `message_type` - (Optional) How the message should be formatted; one of: `classic`, `loggly`, `logplex` or `blank`. Default `classic`. See Fastly's Documentation on Sumologic

The `gcslogging` block supports:

- `name` - (Required) A unique name to identify this GCS endpoint.

- `email` - (Required) The email address associated with the target GCS bucket on your account. You may optionally provide this secret via an environment variable, `FASTLY_GCS_EMAIL`.
- `bucket_name` - (Required) The name of the bucket in which to store the logs.
- `secret_key` - (Required) The secret key associated with the target gcs bucket on your account. You may optionally provide this secret via an environment variable, `FASTLY_GCS_SECRET_KEY`. A typical format for the key is PEM format, containing actual newline characters where required.
- `path` - (Optional) Path to store the files. Must end with a trailing slash. If this field is left empty, the files will be saved in the bucket's root path.
- `period` - (Optional) How frequently the logs should be transferred, in seconds. Default `3600`.
- `gzip_level` - (Optional) Level of GZIP compression, from `0-9`. `0` is no compression. `1` is fastest and least compressed, `9` is slowest and most compressed. Default `0`.
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (`%h %l %u %t %r %>s`)
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.
- `message_type` - (Optional) How the message should be formatted; one of: `classic`, `loggly`, `logplex` or `blank`. Default `classic`. Fastly Documentation

The `syslog` block supports:

- `name` - (Required) A unique name to identify this Syslog endpoint.
- `address` - (Required) A hostname or IPv4 address of the Syslog endpoint.
- `port` - (Optional) The port associated with the address where the Syslog endpoint can be accessed. Default `514`.
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (%h %l %u %t %r %>s)
- `format_version` - (Optional) The version of the custom logging format used for the configured endpoint. Can be either 1 (the default, version 1 log format) or 2 (the version 2 log format).
- `token` - (Optional) Whether to prepend each message with a specific token.
- `use_tls` - (Optional) Whether to use TLS for secure logging. Default `false`.
- `tls_hostname` - (Optional) Used during the TLS handshake to validate the certificate.
- `tls_ca_cert` - (Optional) A secure certificate to authenticate the server with.
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed informa-

tion about Conditionals, see Fastly's Documentation on Conditionals.

- `message_type` - (Optional) How the message should be formatted; one of: `classic`, `loggly`, `logplex` or `blank`. Default `classic`.

The `logentries` block supports:

- `name` - (Required) A unique name to identify this GCS endpoint.
- `token` - (Required) Logentries Token to be used for authentication (https://logentries.com/doc/input-token/).
- `port` - (Optional) The port number configured in Logentries to send logs to. Defaults to `20000`.
- `use_tls` - (Optional) Whether to use TLS for secure logging. Defaults to `true`
- `format` - (Optional) Apache-style string or VCL variables to use for log formatting. Defaults to Apache Common Log format (`%h %l %u %t %r %>s`).
- `format_version` - (Optional) The version of the custom logging format used for the configured endpoint. Can be either 1 (the default, version 1 log format) or 2 (the version 2 log format).
- `response_condition` - (Optional) Name of already defined `condition` to apply. This `condition` must be of type `RESPONSE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `response_object` block supports:

- `name` - (Required) A unique name to identify this Response Object.
- `status` - (Optional) The HTTP Status Code. Default `200`.
- `response` - (Optional) The HTTP Response. Default `Ok`.
- `content` - (Optional) The content to deliver for the response object.
- `content_type` - (Optional) The MIME type of the content.
- `request_condition` - (Optional) Name of already defined `condition` to be checked during the request phase. If the condition passes then this object will be delivered. This `condition` must be of type `REQUEST`.
- `cache_condition` - (Optional) Name of already defined `condition` to check after we have retrieved an object. If the condition passes then deliver this Request Object instead. This `condition` must be of type `CACHE`. For detailed information about Conditionals, see Fastly's Documentation on Conditionals.

The `vcl` block supports:

- `name` - (Required) A unique name for this configuration block.
- `content` - (Required) The custom VCL code to upload.
- `main` - (Optional) If `true`, use this block as the main configuration. If `false`, use this block as an includable library. Only a single VCL block can be marked as the main block. Default is `false`.

11

» Attributes Reference

The following attributes are exported:

- `id` - The ID of the Service.
- `name` – Name of this service.
- `active_version` - The currently active version of your Fastly Service.
- `domain` – Set of Domains. See above for details.
- `backend` – Set of Backends. See above for details.
- `header` – Set of Headers. See above for details.
- `s3logging` – Set of S3 Logging configurations. See above for details.
- `papertrail` – Set of Papertrail configurations. See above for details.
- `response_object` - Set of Response Object configurations. See above for details.
- `vcl` – Set of custom VCL configurations. See above for details.
- `default_host` – Default host specified.
- `default_ttl` - Default TTL.
- `force_destroy` - Force the destruction of the Service on delete.