

Middleware Overview



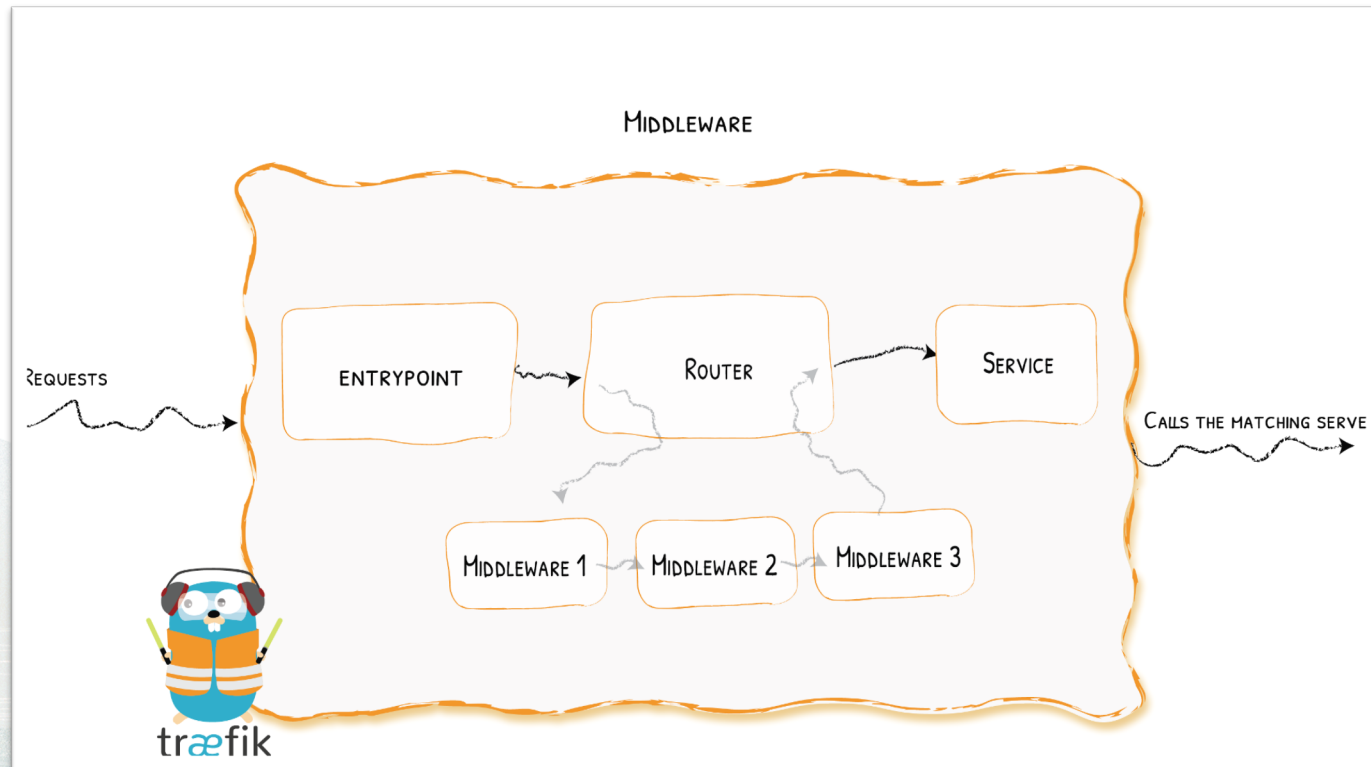


Overview Middleware

- **Middleware connect to Routers** – One or more Middlewares can connect to one or more Routers.
- **Intercepts Requests** – The Middleware transforms requests before they are sent to a service



Middleware





Types of Middlewares

- **Authentication** – Handles user/service authentication
- **Content Modifier** – Manages the served content
- **Middleware Tool** – Middleware Management
- **Path Modifier** – Configures the handling of the URL structure
- **Request Lifecycle** – Applies rules to service requests
- **Security** – Secures access to the service



Available Middleware

Available Middlewares

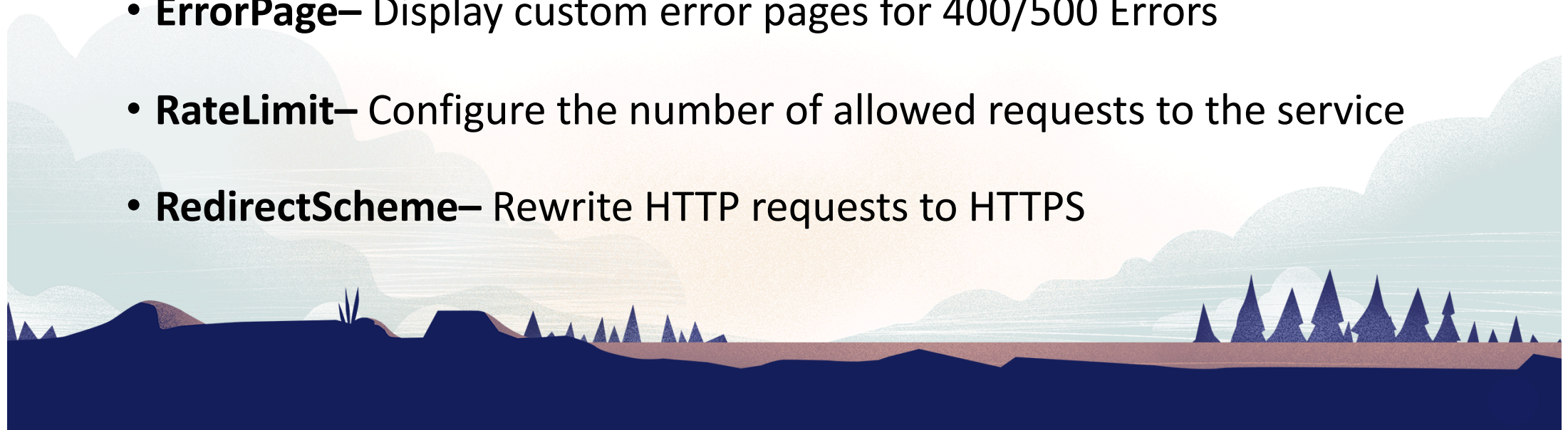
Middleware	Purpose	Area
AddPrefix	Add a Path Prefix	Path Modifier
BasicAuth	Basic auth mechanism	Security, Authentication
Buffering	Buffers the request/response	Request Lifecycle
Chain	Combine multiple pieces of middleware	Middleware tool
CircuitBreaker	Stop calling unhealthy services	Request Lifecycle
Compress	Compress the response	Content Modifier
DigestAuth	Adds Digest Authentication	Security, Authentication
Errors	Define custom error pages	Request Lifecycle
ForwardAuth	Authentication delegation	Security, Authentication
Headers	Add / Update headers	Security

IPWhiteList	Limit the allowed client IPs	Security, Request lifecycle
InFlightReq	Limit the number of simultaneous connections	Security, Request lifecycle
PassTLSClientCert	Adding Client Certificates in a Header	Security
RateLimit	Limit the call frequency	Security, Request lifecycle
RedirectScheme	Redirect easily the client elsewhere	Request lifecycle
RedirectRegex	Redirect the client elsewhere	Request lifecycle
ReplacePath	Change the path of the request	Path Modifier
ReplacePathRegex	Change the path of the request	Path Modifier
Retry	Automatically retry the request in case of errors	Request lifecycle
StripPrefix	Change the path of the request	Path Modifier
StripPrefixRegex	Change the path of the request	Path Modifier



Middleware Labs

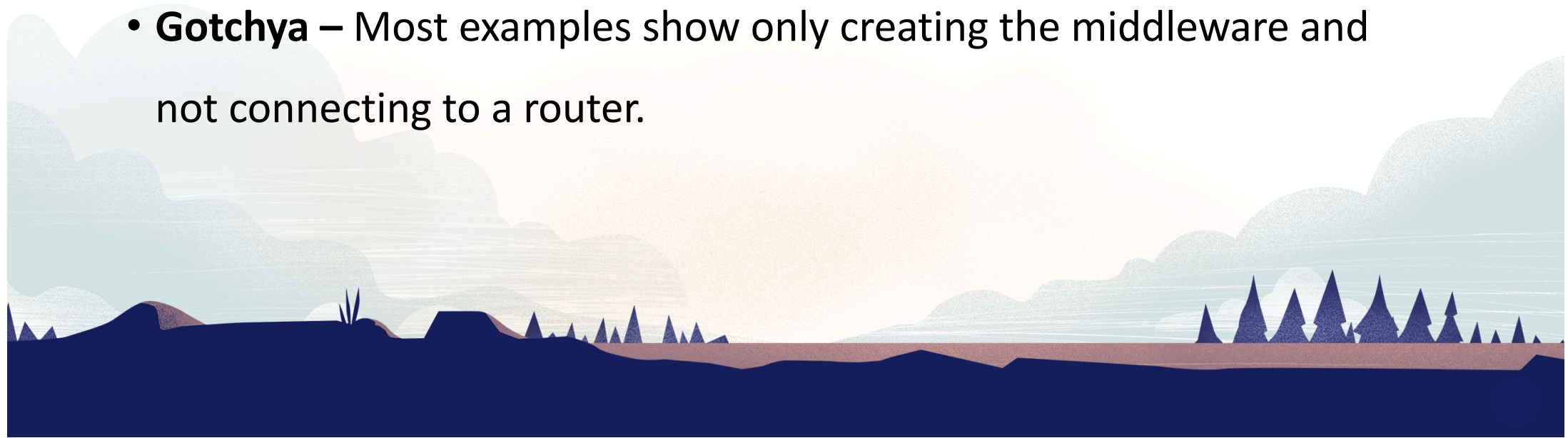
- **BasicAuth** – Add Basic authentication to services
- **Compress**– Enables gzip compression on severed content
- **ErrorPage**– Display custom error pages for 400/500 Errors
- **RateLimit**– Configure the number of allowed requests to the service
- **RedirectScheme**– Rewrite HTTP requests to HTTPS





Middleware Deployment

- **Create Middleware** – Define your middleware
- **Router <-> Middleware** – Connect the Router to the Middleware
- **Gotcha** – Most examples show only creating the middleware and not connecting to a router.





Understanding Middleware Labels

Let's take a look at how the Traefik labels are constructed

traefik **http** **middleware** **<middleware name>**.rule

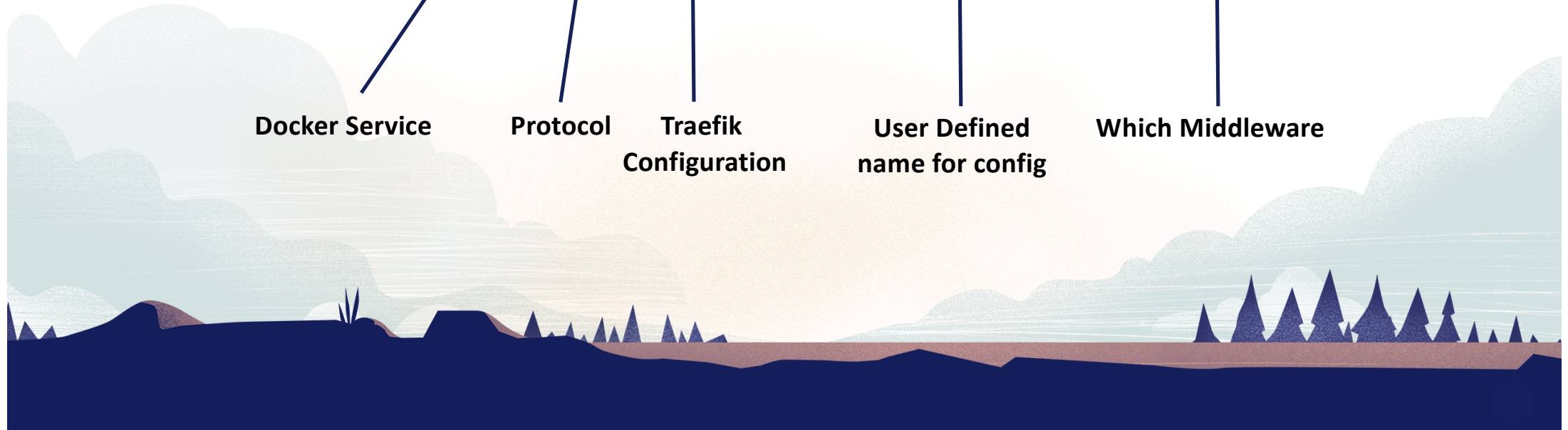
Docker Service

Protocol

Traefik
Configuration

User Defined
name for config

Which Middleware



Middleware Deployment



```
# Add the catapp service
catapp:
  # Random Cat GIF generaor
  image: mikesir87/cats:1.0
  labels:
# Routers
  - "traefik.http.routers.catapp.rule=Host(`catapp.localhost`)"
  - "traefik.http.routers.catapp.service=catapp"
  - "traefik.http.routers.catapp.entrypoints=web"
  - "traefik.http.routers.catapp.middlewares=test-compress"
# Services
  - "traefik.http.services.catapp.loadbalancer.server.port=5000"
# Middlewares
  - "traefik.http.middlewares.test-compress.compress=true"
```