Traefik Chain Middleware Documentation

- Configuration Example
- Configuration Options

The `chain` middleware enables you to define reusable combinations of other pieces of middleware. It makes it effortless to reuse the same groups.

Configuration Example

Below is an example of a Chain containing `AllowList`, `BasicAuth`, and `RedirectScheme`.

```
# ...
http:
  routers:
    router1:
      service: service1
      middlewares:
        - secured
      rule: "Host(`mydomain`)"

  middlewares:
    secured:
      chain:
        middlewares:
          - https-only
          - known-ips
          - auth-users

    auth-users:
```

```
      basicAuth:
        users:
          - "test:$apr1$H6uskkkW$IgXLP6ewTrSuBkTrqE8wj/"

    https-only:
      redirectScheme:
        scheme: https

    known-ips:
      ipAllowList:
        sourceRange:
          - "192.168.1.7"
          - "127.0.0.1/32"

  services:
    service1:
      loadBalancer:
        servers:
          - url: "http://127.0.0.1:80"
```

```
# ...
[http.routers]
  [http.routers.router1]
    service = "service1"
    middlewares = ["secured"]
    rule = "Host(`mydomain`)"

[http.middlewares]
  [http.middlewares.secured.chain]
    middlewares = ["https-only", "known-ips", "auth-users"]

  [http.middlewares.auth-users.basicAuth]
    users = ["test:$apr1$H6uskkkW$IgXLP6ewTrSuBkTrqE8wj/"]

  [http.middlewares.https-only.redirectScheme]
    scheme = "https"

  [http.middlewares.known-ips.ipAllowList]
    sourceRange = ["192.168.1.7", "127.0.0.1/32"]
```

```
[http.services]
  [http.services.service1]
    [http.services.service1.loadBalancer]
      [[http.services.service1.loadBalancer.servers]]
        url = "http://127.0.0.1:80"
```

```
labels:
  - "traefik.http.routers.router1.service=service1"
  - "traefik.http.routers.router1.middlewares=secured"
  - "traefik.http.routers.router1.rule=Host(`mydomain`)"
  - "traefik.http.middlewares.secured.chain.middlewares=https-only,known-ips,auth-users"
  - "traefik.http.middlewares.auth-
users.basicauth.users=test:$apr1$H6uskkkW$IgXLP6ewTrSuBkTrqE8wj/"
  - "traefik.http.middlewares.https-only.redirectscheme.scheme=https"
  - "traefik.http.middlewares.known-ips.ipallowlist.sourceRange=192.168.1.7,127.0.0.1/32"
  - "traefik.http.services.service1.loadbalancer.server.port=80"
```

```
{
  // ...
  "Tags": [
    "traefik.http.routers.router1.service=service1",
    "traefik.http.routers.router1.middlewares=secured",
    "traefik.http.routers.router1.rule=Host(`mydomain`)",
    "traefik.http.middlewares.secured.chain.middlewares=https-only,known-ips,auth-users",
    "traefik.http.middlewares.auth-
users.basicauth.users=test:$apr1$H6uskkkW$IgXLP6ewTrSuBkTrqE8wj/",
    "traefik.http.middlewares.https-only.redirectscheme.scheme=https",
    "traefik.http.middlewares.known-ips.ipallowlist.sourceRange=192.168.1.7,127.0.0.1/32",
    "traefik.http.services.service1.loadbalancer.server.port=80"
  ]
}
```

```
apiVersion: traefik.io/v1alpha1
kind: IngressRoute
metadata:
  name: test
  namespace: default
spec:
  entryPoints:
```

```
    - web
  routes:
    - match: Host(`mydomain`)
      kind: Rule
      services:
        - name: whoami
          port: 80
      middlewares:
        - name: secured
---
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
  name: secured
spec:
  chain:
    middlewares:
    - name: https-only
    - name: known-ips
    - name: auth-users
---
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
  name: auth-users
spec:
  basicAuth:
    users:
    - test:$apr1$H6uskkkW$IgXLP6ewTrSuBkTrqE8wj/
---
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
  name: https-only
spec:
  redirectScheme:
    scheme: https
---
apiVersion: traefik.io/v1alpha1
```

```
kind: Middleware
metadata:
  name: known-ips
spec:
  ipAllowList:
    sourceRange:
    - 192.168.1.7
    - 127.0.0.1/32
```

Configuration Options

Field	Description	Default	Required
`middlewares`	List of middlewares to chain. The middlewares have to be in the same namespace as the `chain` middleware.	[]	Yes

---

□□□ #2
□ □□□ □□□ 2026-10-31 14:02:37 UTC
□ □□□ □□□ 2026-10-31 14:02:37 UTC