

GoFrame<https://github.com/gogf/gf/tree/master/contrib/config>

polaris, apollo, nacos, consul, kubernetes configmap

main polaris <https://github.com/gogf/gf/tree/master/contrib/config/polaris>

boot

```
package boot

import (
    "github.com/gogf/gf/contrib/config/polaris/v2"
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/os/gctx"
)

func init() {
    var (
        ctx      = gctx.GetInitCtx()
        namespace = "default"
        fileGroup = "TestGroup"
        fileName  = "config.yaml"
        path      = "manifest/config/polaris.yaml"
        logDir    = "/tmp/polaris/log"
    )
    // Create polaris Client that implements gcfg.Adapter.
    adapter, err := polaris.New(ctx, polaris.Config{
        Namespace: namespace,
        FileGroup: fileGroup,
        FileName:  fileName,
        Path:      path,
        LogDir:    logDir,
        Watch:     true,
    })
    if err != nil {
        g.Log().Fatalf(ctx, ` %+v`, err)
    }
    // Change the adapter of default configuration instance.
    g.Cfg().SetAdapter(adapter)
}
```

- Namespacepolaris
- FileGrouppolaris
- Filenamepolaris
- Pathpolarispolaris

Polaris

Content Menu

-
-
-
-
- polaris
-
-
-

```

global:
  serverConnector:
    addresses:
      - 127.0.0.1:8091
  config:
    configConnector:
      addresses:
        - 127.0.0.1:8093
  consumer:
    localCache:
      persistDir: "/tmp/polaris/backup"

```

main.goboot

```

package main

import (
    _ "github.com/gogf/gf/example/config/polaris/boot"

    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/os/gctx"
)

func main() {
    var ctx = gctx.GetInitCtx()

    // Available checks.
    g.Dump(g.Cfg().Available(ctx))

    // All key-value configurations.
    g.Dump(g.Cfg().Data(ctx))

    // Retrieve certain value by key.
    g.Dump(g.Cfg().MustGet(ctx, "server.address"))
}

```

import

```

_ "github.com/gogf/gf/example/config/polaris/boot"

```

file		
apollo	https://github.com/gogf/gf/tree/master/contrib/config/apollo	
kubecm	https://github.com/gogf/gf/tree/master/contrib/config/kubecm	
nacos	https://github.com/gogf/gf/tree/master/contrib/config/nacos	
polaris	https://github.com/gogf/gf/tree/master/contrib/config/polaris	
consul	https://github.com/gogf/gf/tree/master/contrib/config/consul	

<https://github.com/gogf/gf/tree/master/contrib/config>

<https://github.com/gogf/gf/tree/master/example/config/polaris>

polaris

```
docker run -d --name polaris -p 8080:8080 -p 8090:8090 -p 8091:8091 -p 8093:8093 loads/polaris-server-standalone:1.11.2
```

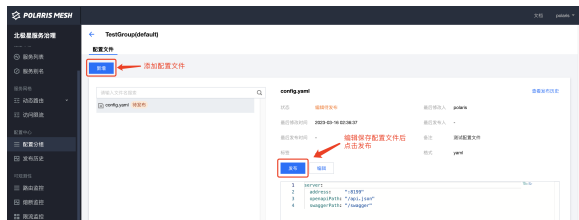
```
$ go run main.go
true
{}
"failed to update local value: config file is empty"
panic: failed to update local value: config file is empty

goroutine 1 [running]:
github.com/gogf/gf/v2/os/gcfg.(*Config).MustGet(0x0?, {0x1c1c4f8?, 0xc0000c2000?, {0x1ac11ad?, 0x0?}, {0x0?, 0xc000002340?, 0xc000064738?}})
/Users/john/Workspace/gogf/gf/os/gcfg/gcfg.go:167 +0x5e
main.main()
/Users/john/Workspace/gogf/gf/example/config/polaris/main.go:20 +0x1b8
```

MustGetpolarisMust*panic

polaris

<http://127.0.0.1:8080/#/login> polaris polaris



```
$ go run main.go
true
{
  "server": {
    "openapiPath": "/api.json",
    "swaggerPath": "/swagger",
    "address": ":8199",
  },
}
<nil>
":8199"
```

polaris

