# Manual Scale Tests for Control Plane and Data Plane

1. Scaling Only Data Plane with one Control Plane

|  |  |  |  |
| --- | --- | --- | --- |
| Number of Data Plane | Coffee Svc | Tea Svc | Errors |
| 50 | Traffic sent successfully | Traffic sent successfully |  |
| 100 | Traffic sent successfully | Traffic sent successfully |  |
| 200 | Traffic sent successfully | Traffic sent successfully |  |
| 300 | Traffic sent successfully | Traffic sent successfully |  |
| 400 | Traffic sent successfully | Traffic sent successfully |  |
| 500 | Traffic sent successfully | Traffic sent successfully |  |
| 600 | Traffic sent successfully | Traffic sent successfully |  |
| 700  | Traffic sent successfully | Traffic sent successfully | Connection dropped once for coffee svc, some performance issues |
| 800 |  |  | Doesn’t scale to 800 NGINX pods, remains 700 |

Error in NGF logs:

{"level":"error","ts":"2025-02-18T22:40:00Z","logger":"nginxUpdater.commandService","msg":"error waiting for connection","error":"context canceled","stacktrace":"github.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent.(\*commandService).Subscribe\n\tgithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/command.go:136\ngithub.com/nginx/agent/v3/api/grpc/mpi/v1.\_CommandService\_Subscribe\_Handler\n\tgithub.com/nginx/agent/v3@v3.0.0-20250120091728-0f0c0e2478aa/api/grpc/mpi/v1/command\_grpc.pb.go:233\ngithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/grpc/interceptor.(\*ContextSetter).Stream.ContextSetter.Stream.func1\n\tgithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/grpc/interceptor/interceptor.go:43\ngoogle.golang.org/grpc.(\*Server).processStreamingRPC\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1689\ngoogle.golang.org/grpc.(\*Server).handleStream\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1806\ngoogle.golang.org/grpc.(\*Server).serveStreams.func2.1\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1030"}

1. Configuring 10 Control Plane with 50 Data plane

|  |  |  |
| --- | --- | --- |
| Control Plane | Data Plane  | Leader Lease |
| ngf-nginx-gateway-fabric-5fc884856b-6wkxz | 1. gateway-nginx-84b7cbf8cd-p4sfw
2. gateway-nginx-84b7cbf8cd-299vs
3. gateway-nginx-84b7cbf8cd-xx97k
4. gateway-nginx-84b7cbf8cd-b7g8p
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-cw6mz | 1. gateway-nginx-84b7cbf8cd-km7fs
2. gateway-nginx-84b7cbf8cd-qzrsm
3. gateway-nginx-84b7cbf8cd-2kxhc
4. gateway-nginx-84b7cbf8cd-9bvhb
5. gateway-nginx-84b7cbf8cd-v86jc
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-dlvtk | 1. gateway-nginx-84b7cbf8cd-b8x9c
2. gateway-nginx-84b7cbf8cd-b5f69
3. gateway-nginx-84b7cbf8cd-js8qg
4. gateway-nginx-84b7cbf8cd-qbtpp
5. gateway-nginx-84b7cbf8cd-vlt5p
6. gateway-nginx-84b7cbf8cd-c2sn5
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-fpdvq | 1. gateway-nginx-84b7cbf8cd-xph9x
2. gateway-nginx-84b7cbf8cd-l7f2k
3. gateway-nginx-84b7cbf8cd-qtbgd
4. gateway-nginx-84b7cbf8cd-8cxtg
5. gateway-nginx-84b7cbf8cd-5z6bs
6. gateway-nginx-84b7cbf8cd-rpqqx
7. gateway-nginx-84b7cbf8cd-dwlnh
8. gateway-nginx-84b7cbf8cd-tf8r5
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-ghgz7 | 1. gateway-nginx-84b7cbf8cd-x7hcp
2. gateway-nginx-84b7cbf8cd-jf8wk
3. gateway-nginx-84b7cbf8cd-wwtsg
4. gateway-nginx-84b7cbf8cd-ccxs5
5. gateway-nginx-84b7cbf8cd-9vmht
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-n58j9 | 1. gateway-nginx-84b7cbf8cd-lmv62
2. gateway-nginx-84b7cbf8cd-nhx2q
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-r9pln | 1. gateway-nginx-84b7cbf8cd-5c8ck
2. gateway-nginx-84b7cbf8cd-ckdnv
3. gateway-nginx-84b7cbf8cd-bqvtj
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-rxtz8 | 1. gateway-nginx-84b7cbf8cd-sq275
2. gateway-nginx-84b7cbf8cd-mhfq6
 |  |
| ngf-nginx-gateway-fabric-5fc884856b-vcgkx | 1. gateway-nginx-84b7cbf8cd-2lsss
2. gateway-nginx-84b7cbf8cd-gnl6z
3. gateway-nginx-84b7cbf8cd-h299k
4. gateway-nginx-84b7cbf8cd-tfc7n
5. gateway-nginx-84b7cbf8cd-fq86v
6. gateway-nginx-84b7cbf8cd-c96mm
7. gateway-nginx-84b7cbf8cd-js5q4
8. gateway-nginx-84b7cbf8cd-pg825
9. gateway-nginx-84b7cbf8cd-rwgzl
10. gateway-nginx-84b7cbf8cd-sklpl
 | Acquired Leader lease |
| ngf-nginx-gateway-fabric-5fc884856b-vg7jd | 1. gateway-nginx-84b7cbf8cd-mfvb4
2. gateway-nginx-84b7cbf8cd-6tlsz
3. gateway-nginx-84b7cbf8cd-6qdnn
4. gateway-nginx-84b7cbf8cd-ztx44
5. gateway-nginx-84b7cbf8cd-dcc9s
 |  |

1. Configuring 20 control planes with 100 data planes
	1. Traffic sent successfully to coffee, and tea
	2. Did not check config being correctly applied against 100 instances
	3. All NGINX pods running
2. Configuring 20 control planes with 300 data planes
	1. Did not check config being correctly applied against 300 instances
	2. Traffic sent successfully to coffee, and tea
	3. All NGINX pods running
3. Configuring 20 control planes with 500 data planes
	1. Did not check config being correctly applied against 500 instances
	2. Traffic sent successfully to coffee, and tea
	3. All NGINX pods running
4. Configuring 20 control planes with 700 data planes
	1. Did not check config being correctly applied against 500 instances
	2. Traffic sent successfully to coffee, and tea
	3. All NGINX pods running
5. Configuring 20 control planes with 800 data planes
	1. Traffic sent successfully to coffee, and tea
	2. All NGINX pods running
6. Configuring 20 control planes with 1000 data planes
	1. Traffic sent successfully to coffee, and tea
	2. Only 913 pods come to running state
	3. It then drops to 613 pods running
	4. All pods terminated, checked logs for pod in error state.
	5. Control plane 3 pods restart and recover but not the same for NGINX pods
	6. **Errors** in control plane logs about not able to get pod owner
	el":"info","ts":"2025-02-19T20:24:46Z","logger":"nginxUpdater.commandService","msg":"Creating connection for nginx pod: gateway-nginx-84b7cbf8cd-9wqfx"}

{"level":"error","ts":"2025-02-19T20:24:46Z","logger":"nginxUpdater.commandService","msg":"error getting pod owner","error":"failed to get nginx Pod's ReplicaSet: replicasets.apps \"gateway-nginx-84b7cbf8cd\" not found","stacktrace":"github.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent.(\*commandService).CreateConnection\n\tgithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/command.go:96\ngithub.com/nginx/agent/v3/api/grpc/mpi/v1.\_CommandService\_CreateConnection\_Handler.func1\n\tgithub.com/nginx/agent/v3@v3.0.0-20250120091728-0f0c0e2478aa/api/grpc/mpi/v1/command\_grpc.pb.go:191\ngithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/grpc/interceptor.(\*ContextSetter).Unary.ContextSetter.Unary.func1\n\tgithub.com/nginx/nginx-gateway-fabric/internal/mode/static/nginx/agent/grpc/interceptor/interceptor.go:60\ngithub.com/nginx/agent/v3/api/grpc/mpi/v1.\_CommandService\_CreateConnection\_Handler\n\tgithub.com/nginx/agent/v3@v3.0.0-20250120091728-0f0c0e2478aa/api/grpc/mpi/v1/command\_grpc.pb.go:193\ngoogle.golang.org/grpc.(\*Server).processUnaryRPC\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1392\ngoogle.golang.org/grpc.(\*Server).handleStream\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1802\ngoogle.golang.org/grpc.(\*Server).serveStreams.func2.1\n\tgoogle.golang.org/grpc@v1.69.4/server.go:1030"}

* 1. Data plane pods never restarted, all terminated.

**Errors**

1. Pod **ngf-nginx-gateway-fabric-5fc884856b-dlvtk** failed to leader election, lock mechanism failed

{"level":"error","ts":"2025-02-19T15:43:53Z","msg":"Failed to update lock: Operation cannot be fulfilled on leases.coordination.k8s.io \"ngf-nginx-gateway-fabric-leader-election\": the object has been modified; please apply your changes to the latest version and try again","stacktrace":"k8s.io/client-go/tools/leaderelection.(\*LeaderElector).tryAcquireOrRenew\n\tk8s.io/client-go@v0.32.1/tools/leaderelection/leaderelection.go:472\nk8s.io/client-go/tools/leaderelection.(\*LeaderElector).acquire.func1\n\tk8s.io/client-go@v0.32.1/tools/leaderelection/leaderelection.go:260\nk8s.io/apimachinery/pkg/util/wait.BackoffUntil.func1\n\tk8s.io/apimachinery@v0.32.1/pkg/util/wait/backoff.go:226\nk8s.io/apimachinery/pkg/util/wait.BackoffUntil\n\tk8s.io/apimachinery@v0.32.1/pkg/util/wait/backoff.go:227\nk8s.io/apimachinery/pkg/util/wait.JitterUntil\n\tk8s.io/apimachinery@v0.32.1/pkg/util/wait/backoff.go:204\nk8s.io/client-go/tools/leaderelection.(\*LeaderElector).acquire\n\tk8s.io/client-go@v0.32.1/tools/leaderelection/leaderelection.go:258\nk8s.io/client-go/tools/leaderelection.(\*LeaderElector).Run\n\tk8s.io/client-go@v0.32.1/tools/leaderelection/leaderelection.go:215\nsigs.k8s.io/controller-runtime/pkg/manager.(\*controllerManager).Start.func3\n\tsigs.k8s.io/controller-runtime@v0.20.1/pkg/manager/internal.go:449"}

2. Pod **ngf-nginx-gateway-fabric-5fc884856b-vg7jd**

{"level":"info","ts":"2025-02-19T15:44:48Z","logger":"eventLoop.eventHandler","msg":"Handling events didn't result into NGINX configuration changes","batchID":38}

{"level":"info","ts":"2025-02-19T15:44:48Z","msg":"k8s.io/client-go@v0.32.1/tools/cache/reflector.go:251: watch of \*v1.EndpointSlice ended with: an error on the server (\"unable to decode an event from the watch stream: got short buffer with n=0, base=4092, cap=20480\") has prevented the request from succeeding"}

**Observations**

* Nginx Instances always connect to a NGF instances
* Leader lease is successfully acquired by one of the NGF instances
* With one control plane, scaling stops after 700 nginx instances. Still traffic could be sent to applications even if not scaling.
* With 20 control plane and 1000 nginx instances, all NGINX instance crashes, and some control plane pods restart due to pod ownership errors for NGINX pods.