

Set the Number of Reconnect Attempts

Applications can set the maximum reconnect attempts per server. This includes the server provided to the client's connect call, as well as the server the client discovered through another server. Once reconnect to a server fails the specified amount of times in a row, it will be removed from the connect list. After a successful reconnect to a server, the client will reset that server's failed reconnect attempt count. If a server was removed from the connect list, it can be rediscovered on connect. This effectively resets the connect attempt count as well. If the client runs out of servers to reconnect, it will close the connection and [raise an error](#).

Go

```
// Set max reconnects attempts
nc, err := nats.Connect("demo.nats.io", nats.MaxReconnects(10))
if err != nil {
    log.Fatal(err)
}
defer nc.Close()

// Do something with the connection
```

Java

```
Options options = new Options.Builder()
    .server("nats://demo.nats.io:4222")
    .maxReconnects(10) // Set max reconnect attempts
    .build();
Connection nc = Nats.connect(options);

// Do something with the connection

nc.close();
```

JavaScript

```
const nc = await connect({
  maxReconnectAttempts: 10,
  servers: ["demo.nats.io"],
});
```

Python

```
nc = NATS()
await nc.connect(
  servers=["nats://demo.nats.io:4222"],
  max_reconnect_attempts=10,
)

# Do something with the connection

await nc.close()
```

C#

```
// dotnet add package NATS.Net
using NATS.Net;
using NATS.Client.Core;

await using var client = new NatsClient(new NatsOpts
{
  Url = "nats://demo.nats.io:4222",

  // Set the maximum number of reconnect attempts
  MaxReconnectRetry = 10,
});
```

Ruby

```
require 'nats/client'

NATS.start(servers: ["nats://127.0.0.1:1222", "nats://127.0.0.1:1223",
"nats://127.0.0.1:1224"], max_reconnect_attempts: 10) do |nc|
  # Do something with the connection

  # Close the connection
  nc.close
```

```
end
```

C

```
natsConnection      *conn    = NULL;
natsOptions          *opts    = NULL;
natsStatus           s        = NATS_OK;

s = natsOptions_Create(&opts);
if (s == NATS_OK)
    s = natsOptions_SetMaxReconnect(opts, 10);
if (s == NATS_OK)
    s = natsConnection_Connect(&conn, opts);

(...)

// Destroy objects that were created
natsConnection_Destroy(conn);
natsOptions_Destroy(opts);
```

📄 #3

📄 cocobook-sync 📄 2026-10-28 11:12:27 UTC

📄 cocobook-sync 📄 2026-10-29 01:42:36 UTC