

socks

Migrating from v1

For the most part, migrating from v1 takes minimal effort as v2 still supports factory creation of proxy connections with callback support.

Notable breaking changes

- In an options object, the proxy 'command' is now required and does not default to 'connect'.
- **In an options object, 'target' is now known as 'destination'.**
- Sockets are no longer paused after a SOCKS connection is made, so `socket.resume()` is no longer required. (Please be sure to attach data handlers immediately to the Socket to avoid losing data).
- In v2, only the 'connect' command is supported via the factory `SocksClient.createConnection` function. (BIND and ASSOCIATE must be used with a `SocksClient` instance via event handlers).
- In v2, the factory `SocksClient.createConnection` function callback is called with a single object rather than separate socket and info object.
- A SOCKS http/https agent is no longer bundled into the library.

For informational purposes, here is the original getting started example from v1 converted to work with v2.

Before (v1)

```
var Socks = require('socks');

var options = {
  proxy: {
    ipAddress: "202.101.228.108",
    port: 1080,
    type: 5
  },
  target: {
    host: "google.com",
    port: 80
  },
  command: 'connect'
};

Socks.createConnection(options, function(err, socket, info) {
  if (err)
    console.log(err);
  else {
    socket.write("GET / HTTP/1.1\nHost: google.com\n\n");
    socket.on('data', function(data) {
      console.log(data.length);
      console.log(data);
    });

    // PLEASE NOTE: sockets need to be resumed before any data will come in or
    out as they are paused right before this callback is fired.
    socket.resume();
  }
});
```

```

        // 569
        // <Buffer 48 54 54 50 2f 31 2e 31 20 33 30 31 20 4d 6f 76 65 64 20 50 65...
    }
});

```

After (v2)

```

const SocksClient = require('socks').SocksClient;

let options = {
  proxy: {
    ipaddress: "202.101.228.108",
    port: 1080,
    type: 5
  },
  destination: {
    host: "google.com",
    port: 80
  },
  command: 'connect'
};

SocksClient.createConnection(options, function(err, result) {
  if (err)
    console.log(err);
  else {
    result.socket.write("GET / HTTP/1.1\nHost: google.com\n\n");
    result.socket.on('data', function(data) {
      console.log(data.length);
      console.log(data);
    });

    // 569
    // <Buffer 48 54 54 50 2f 31 2e 31 20 33 30 31 20 4d 6f 76 65 64 20 50 65...
  }
});

```