spiral Documentation

Release 0.14.1.0

Aaron Gallagher

April 12, 2014

Contents

A twisted curve is a spiral.

`spiral` is a collection of elliptic-curve-backed protocol implementations. At the moment, this is limited to DNSCurve and CurveCP.

Note: `spiral` is still experimental in general; no guarantees are made about the continued existence of APIs.

CHAPTER 1

DNSCurve

DNSCurve support is experimental and requires a currently-unmerged branch of Twisted. Client recursive and nonrecursive resolvers have been implemented; there is currently no DNSCurve server support.

CHAPTER 2

CurveCP

CurveCP support comes in two forms:

2.1 `curvecpmclient` and `curvecpmserver`

`curvecpmclient` and `curvecpmserver` are UCSPI-style executables. `curvecpmclient` will connect to a given CurveCP server and spawn a process to communicate with it. `curvecpmserver` will listen on a particular port and spawn a process for each incoming connection.

2.2 Endpoints

Two standard twisted endpoints are exposed for writing clients or servers in python: `CurveCPClientEndpoint` and `CurveCPServerEndpoint`.

CHAPTER 3

API

3.1 spiral.curvecp

class spiral.curvecp.**CurveCPClientEndpoint**(*reactor*, *host*, *port*, *serverKey*, *serverExtension='\x00' * 16*, *clientKey=None*, *clientExtension='\x00' * 16*)

An IStreamClientEndpoint implementer for CurveCP clients.

Parameters

- **reactor** – An IReactorUDP and IReactorTime provider.
- **host** – A hostname to connect to.
- **port** – The port to connect to the *host* on.
- **serverKey** – The server's public key, as a 32-byte string.
- **serverExtension** – Optionally, the 16-byte server extension. Defaults to all null bytes.
- **clientKey** – Optionally, an `IKeyAndNonceScheme` provider for the client's private key. Defaults to generating an ephemeral key for client the on every new connection.
- **clientExtension** – Optionally, the 16-byte client extension. Defaults to all null bytes.

connect(*fac*)

Connect to a CurveCP host.

Parameters **fac** – An IProtocolFactory provider whose `buildProtocol` returns an IHalfCloseableProtocol provider.

class spiral.curvecp.**CurveCPServerEndpoint**(*reactor*, *serverKey*, *port*)

An IStreamServerEndpoint implementer for CurveCP servers.

All incoming connections are accepted; for filtering based on client/server extension or server DNS name, please override `buildProtocol` to make decisions about the `ICurveCPAddress` provider passed to it.

Parameters

- **reactor** – An IReactorUDP and IReactorTime provider.
- **port** – The port to listen on.
- **serverKey** – An `IKeyAndNonceScheme` provider representing the server's private key.

listen(*fac*)

Listen for incoming CurveCP connections.

Parameters **fac** – An IProtocolFactory provider whose `buildProtocol` returns an IHalfCloseableProtocol provider.

interface `spiral.curvecp.address.`**`ICurveCPAddress`**

`clientExtension` **= <zope.interface.interface.Attribute object at 0x236ae50>**
The 16-byte client extension associated with the connection.

`serverDomain` **= <zope.interface.interface.Attribute object at 0x236af10>**
A string representing the server's DNS name.

`serverExtension` **= <zope.interface.interface.Attribute object at 0x236aed0>**
The 16-byte server extension associated with the connection.

`longTermKey` **= <zope.interface.interface.Attribute object at 0x236af50>**
A `nacl.public.PublicKey` representing the other side's long term public key.

`transportHost` **= <zope.interface.interface.Attribute object at 0x236af90>**
The host or IP of the other side of this connection.

`transportPort` **= <zope.interface.interface.Attribute object at 0x236afd0>**
The port of the other side of this connection.

3.2 spiral.keys

interface `spiral.keys.`**`IKeyAndNonceScheme`**
A key and nonce generation scheme.

`nonce`(*longterm=False*)
Generate a nonce.

Parameters **longterm** – True to increment the long-term counter; False to increment the short-term counter.

Returns 16 bytes.

`key` **= <zope.interface.interface.Attribute object at 0x23db5d0>**
A nacl.public.PrivateKey instance.

class `spiral.keys.`**`Keydir`**(*keydir*)
A key loaded from disk, probably generated by `curvecpmakekey`.

Nonces are eight random bytes concatenated to a counter persisted to disk.

Parameters **keydir** – The path to a key directory.

class `spiral.keys.`**`EphemeralKey`**
An unpersisted, randomly-generated key.

Nonces are 16 random bytes.

S