Go Documentation Server ▽

Package nonce

```
import "go.timothygu.me/downtomeet/server/impl/nonce"
```

Overview
Index

Overview ▾

Package nonce implements a simple ASCII nonce generator, as well as some randomization utilities.

Index ▾

func NewCryptoRandSource() mathrand.Source64
func RandomBase64(r *rand.Rand, n int) string
type Generator
    func NewGenerator(src rand.Source) *Generator
    func (g *Generator) NewBase64(stateLen int) string

Package files

crypto.go nonce.go rand.go

func **NewCryptoRandSource**

```
func NewCryptoRandSource() mathrand.Source64
```

NewCryptoRandSource returns a rand.Source64 backed by the cryptographically-secure random number generator in crypto/rand. If crypto/rand is unavailable, NewCryptoRandSource panics.

The returned rand.Source64 does not support seeding, and the Seed method is a no-op.

func **RandomBase64**

```
func RandomBase64(r *rand.Rand, n int) string
```

RandomBase64 returns a string of length n, consisting of random characters drawn from the provided rand.Rand. The output string will solely consist of base64url characters; that is, all alphanumerics with - and _ added.

type **Generator**

Generator creates randomly generated ASCII nonces. The set of possible characters in generated nonces is the same as RandomBase64. A Generator instance is safe for concurrent use.

```
type Generator struct {
    // contains filtered or unexported fields
}
```

func NewGenerator

```
func NewGenerator(src rand.Source) *Generator
```

NewGenerator creates a new Generator from the given rand.Source.

func (*Generator) NewBase64

```
func (g *Generator) NewBase64(stateLen int) string
```

NewBase64 creates a new nonce with the given length.