

etag

Create simple HTTP ETags

This module generates HTTP ETags (as defined in RFC 7232) for use in HTTP responses.

Installation

This is a [Node.js](#) module available through the [npm registry](#). Installation is done using the [`npm install` command](#):

API

\$ npm test

Testing

which case it is true.

The default value is false, unless the entity is fs.Stats, in mark (that is, the leading W/). The actual entity tag is the same. Specifies if the generated ETag will include the weak validator

weak

etag accepts these properties in the options object.

Options

res.setHeader('ETag', etag(body))

overwritten by options.weak).

fs.Stats, which will generate a weak ETag (this can be accepted. By default, a strong ETag is generated except for complete body of the entity. Strings, Buffers, and fs.Stats are accepted. By default, a strong ETag is generated except for

etag(entity, options)

```
real - weak x 867,613 ops/sec
  ±0.39% (190 runs sampled)
fake - strong x 401,051 ops/sec
  ±0.40% (189 runs sampled)
fake - weak x 400,100 ops/sec
  ±0.47% (188 runs sampled)
```

License

[MIT](#)

Benchmark

```
$ npm run-script bench
> etag@1.8.1 bench nodejs-etag
> node benchmark/index.js

http_parser@2.7.0
node@6.11.1
v8@5.1.281.103
uv@1.11.0
zlib@1.2.11
ares@1.10.1-DEV
icu@58.2
modules@48
openssl@1.0.2k

> node benchmark/body0-100b.js

100B body

4 tests completed.

buffer - strong x 258,647 ops/sec
  ±1.07% (180 runs sampled)
buffer - weak x 263,812 ops/sec
  ±0.61% (184 runs sampled)
string - strong x 259,955 ops/sec
  ±1.19% (185 runs sampled)
string - weak x 264,356 ops/sec
  ±1.09% (184 runs sampled)

> node benchmark/body1-1kb.js
```

```

1KB body
    > node benchmark/body1.js
        4 tests completed.

        buffer - strong x 55,989 ops/sec
        buffer - weak   x 56,148 ops/sec
        string - strong x 27,345 ops/sec
        string - weak   x 27,496 ops/sec
        buffer - strong x 190,586 ops/sec
        buffer - weak   x 189,018 ops/sec
        string - strong x 188 runs sampled
        string - weak   x 188 runs sampled
        buffer - strong x 182 runs sampled
        buffer - weak   x 182 runs sampled
        string - strong x 144,272 ops/sec
        string - weak   x 145,380 ops/sec
        buffer - strong x 188 runs sampled
        buffer - weak   x 187 runs sampled
        string - strong x 143% (187 runs sampled)
        string - weak   x 143% (187 runs sampled)

5KB body
    > node benchmark/body2-5kb.js
        4 tests completed.

        buffer - strong x 7,083 ops/sec
        buffer - weak   x 7,115 ops/sec
        string - strong x 3,068 ops/sec
        string - weak   x 3,096 ops/sec
        buffer - strong x 92,435 ops/sec
        buffer - weak   x 92,373 ops/sec
        string - strong x 48,850 ops/sec
        string - weak   x 49,380 ops/sec
        buffer - strong x 186 runs sampled
        buffer - weak   x 186 runs sampled
        string - strong x 189 runs sampled
        string - weak   x 190 runs sampled
        buffer - strong x 190 runs sampled
        buffer - weak   x 191 runs sampled
        string - strong x 3,068 ops/sec
        string - weak   x 3,096 ops/sec
        buffer - strong x 92,373 ops/sec
        buffer - weak   x 92,435 ops/sec
        string - strong x 48,850 ops/sec
        string - weak   x 49,380 ops/sec
        buffer - strong x 186 runs sampled
        buffer - weak   x 187 runs sampled
        string - strong x 143% (187 runs sampled)
        string - weak   x 143% (187 runs sampled)

10KB body
    > node benchmark/body3-10kb.js
        4 tests completed.

        buffer - strong x 871,642 ops/sec
        buffer - weak   x 871,642 ops/sec
        string - strong x 189 runs sampled
        string - weak   x 190 runs sampled
        buffer - strong x 188 runs sampled
        buffer - weak   x 189 runs sampled
        string - strong x 48,850 ops/sec
        string - weak   x 49,380 ops/sec
        buffer - strong x 92,435 ops/sec
        buffer - weak   x 92,373 ops/sec
        string - strong x 3,068 ops/sec
        string - weak   x 3,096 ops/sec
        buffer - strong x 190 runs sampled
        buffer - weak   x 191 runs sampled
        string - strong x 3,068 ops/sec
        string - weak   x 3,096 ops/sec
        buffer - strong x 92,373 ops/sec
        buffer - weak   x 92,435 ops/sec
        string - strong x 48,850 ops/sec
        string - weak   x 49,380 ops/sec
        buffer - strong x 186 runs sampled
        buffer - weak   x 187 runs sampled
        string - strong x 143% (187 runs sampled)
        string - weak   x 143% (187 runs sampled)

```

4 tests completed.

10KB body

```
< node_benchmark/body3-10kb.js
```

4 tests completed.
buffer - strong x 92,435 ops/sec
buffer - weak x 98,42% (188 runs sampled)
buffer - weak x 92,373 ops/sec
string - strong x 48,850 ops/sec
string - strong x 48,58% (189 runs sampled)
string - weak x 48,850 ops/sec
string - weak x 49,380 ops/sec
string - weak x 49,56% (190 runs sampled)

4 tests completed.

5KB body

```
< node_benchmark/body>-5kb.js
```

4 tests completed.
buffer - strong x 189,018 ops/sec
buffer - weak x 190,586 ops/sec
buffer - weak (182 runs sampled)
buffer - weak (186 runs sampled)
string - strong x 144,272 ops/sec
string - weak x 145,380 ops/sec
string - weak (188 runs sampled)
string - weak (196 runs sampled)
±1.12% (187 runs sampled)

4 tests completed.