Posted in: Development (/spaces/127/sandpiper/forums/5366/development)

Payload Storage

Doug Winsby (https://autocare.communifire.com/people/dougwinsby)
1/13/2020

We're working on encoding payload data in the `grain` (as stored in the database and transmitted over the wire).

After implementing a [text] -> [gzip] -> [base64] approach, it's interesting to see that small inputs actually explode in length. This makes perfect sense considering the overhead of storing a dictionary in the gzip phase, and the fact that base64 is only 80% efficient (3 bytes become 4).

For example, the first line is encoded as the second line below:

```
sandpiper rocks!
H4sIAAAAAAAAC/ypOzEspyCxILVIoyk/OLlYEAAAA//8BAAD//451mN4QAAAA
```

So 17 characters become 61 characters!

As a simple test, we used our encoding function against different lengths of random text ("Lorem ipsum"), and the "tipping point" is ~260 characters.

Our expected inputs will probably be less random than this test, but it shows we might want different encodings based on the input (either size or type).

For this reason, we're adding an "Encoding Method" enumeration to the grain. This has the added benefit of letting us change our payload encoding in the future as needs change.

https://en.wikipedia.org/wiki/Binary-to-text_encoding (https://en.wikipedia.org/wiki/Binary-to-text_encoding)

Our "grains" table schema now looks like this:

```
CREATE TABLE IF NOT EXISTS "grains" (
  "id"           uuid PRIMARY KEY,
  "slice_id"     uuid REFERENCES "slices" ON DELETE CASCADE,
  "grain_type"   smallint,
  "grain_key"    text,
  "encoding"     smallint,
  "payload"      bytea,
  "created_at"   timestamp,
  CONSTRAINT "grain_type_key" UNIQUE("slice_id", "grain_type", "grain_key")
);
```

Encodings we support initially will be:

```
1: raw
2: base64
3: gzip_base64
```

payload-encoding (/spaces/127/sandpiper/searchresults?keyword=%22payload-encoding%22&searchtags=1)

Like

Reply (/forums/post?tid=5522&ReplyPostID=5523&SpaceID=127)

Doug Winsby (https://autocare.communifire.com/people/dougwinsby)
8/11/2020

We ended up with the following payload encodings:

```
raw: no encoding
a85: ascii85 encoding
b64: base64 encoding
z64: gzip + base64 encoding
z85: gzip + ascii85 encoding
```

Like

Reply (/forums/post?tid=5522&ReplyPostID=6098&SpaceID=127) Answer