

# Bash for Loop Over JSON Array Using jq (/blog/bash-for-loop-over-json-array-using-jq/)

Sometimes you just want to read a JSON config file from Bash and iterate over an array. For example, [when seeding some credentials to a rkandwayne.com\) credential store \(https://github.com/cloudfoundry-community/credhub-seeder-boshrelease/blob/797228c3bf13c0c4f5aefaafa9e38f393bd78b1f/jobs/credhub-seeder/templates/bin/credhub-seeder\)](https://github.com/cloudfoundry-community/credhub-seeder-boshrelease/blob/797228c3bf13c0c4f5aefaafa9e38f393bd78b1f/jobs/credhub-seeder/templates/bin/credhub-seeder). This sometimes can be tricky especially when the JSON contains multi-line strings (for example certificates).

In this blog post I will explain how this can be done with [jq \(https://stedolan.github.io/jq/download/\)](https://stedolan.github.io/jq/download/) and a Bash for loop.

First we will start with some data:

```
sample='[{"name":"foo"}, {"name":"bar"}]'\necho "${sample}" | jq '.'
```

```
[
  {
    "name": "foo"
  },
  {
    "name": "bar"
  }
]
```

By using `jq --compact-output (or -c)` we can get each object on a newline.

```
sample='[{"name":"foo"}, {"name":"bar"}]'\necho "${sample}" | jq -c '.[]'
```

```
{"name":"foo"}\n{"name":"bar"}
```

We could start iterating of the above with a Bash for loop if our data does not contain spaces or newlines. But since certificates contain newlines we better base64 encode each line. Also, instead of `-c`, we now use `-r` to get rid of the extra quotes.


```
sample='[{"name":"foo"}, {"name":"bar"}]'\necho "${sample}" | jq -r '.[] | @base64'
```

```
eyJuYW1lIjoiZm9vIn0=\neyJuYW1lIjoiYmFyIn0=
```

Now let's build our for loop.

```
sample='[{"name":"foo"}, {"name":"bar"}]'\nfor row in $(echo "${sample}" | jq -r '.[] | @base64'); do\n    _jq() {\n        echo ${row} | base64 --decode | jq -r ${1}\n    }\n\n    echo $( _jq '.name' )\ndone
```

```
foo\nbar
```



WRITTEN BY

**Ruben Koster ()**

*Open Source lover with a passion for*





*Cloud Automation.*


*Netherlands*

**Published on**

*July 06, 2017*

SPREAD THE WORD






Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?


Name



**aleksmeshkov** • 2 months ago

Thank you!


^ | v • Reply • Share ›



**Anton Ustyuzhanin** • 3 months ago

Jeez, how brilliant is this! Thank you!


^ | v • Reply • Share ›



**Alexander Lebedev** • 3 months ago

Thanks a lot, incredibly helpful


^ | v • Reply • Share ›



**NaturaGIS** • 5 months ago

Thanks!

^ | v • Reply • Share ›



**Dmitry Kovalevsky** • 6 months ago

Thank you a lot.

This is an easiest way to avoid coding in perl/python/groovy :-) the same.


^ | v • Reply • Share ›



**NikhilO** • 7 months ago

This is great!

^ | v • Reply • Share ›




**Josef Sterntaler** • 7 months ago

I think you can use

```
jq -r '.[]["name"]' <<<$sample
```

instead.


^ | v • Reply • Share ›



**Santosh Kumar Arjunan** • 8 months ago

very good post.. useful information.. thanks..

^ | v • Reply • Share ›



**Acep Saepudin** • 9 months ago


nice! thanks!

^ | v • Reply • Share ›

ALSO ON STARK & WAYNE

How to Reload Buildpacks in CF

1 comment • 4 months ago



**ericpromislow** — Or you could copy `post-start` to a new file called something like `install-buildpacks`, set it to the above code, run it, ...

A Handy S3 CLI


2 comments • 9 months ago



**James Hunt** — Thanks!

Knative on Google with public traffic routing


2 comments • 2 months ago



**Dr Nic Williams** — Try the next article <https://starkandwayne.com/b...>Or do you mean something else by “clean URL”?

ELB & Instance Information From AWS CLI with IAM Profile

1 comment • a year ago



**Srinivasa Katta** — Thanks Chris,your explanation saves a lot of time to understand AWS CLI. Thanks.

