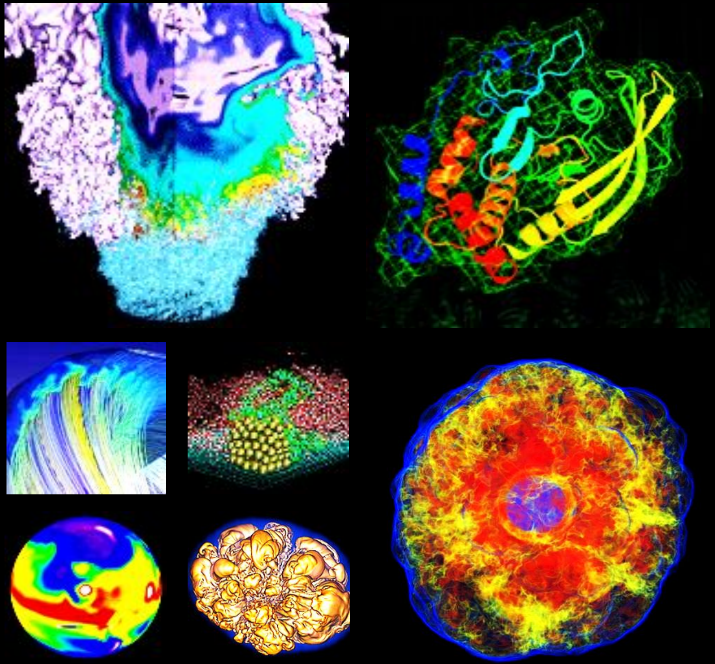
# SSHProxy

## A NUG Joint



Abe Singer

2019-01-24

# sshproxy

Serves time-limited ssh keys

Uses the magic of ssh *certificates*

- ssh keys signed by an *ssh CA cert*
- ssh certs include expiration dates
    - and other restrictions

sshproxy generates ssh key & certificate which user downloads

- Requires MFA
- Default key lifetime: 24 hours
    - Process available for longer keys, with authorization
- RESTful api
- Client-side scripts available to provide simple user interface

# Using sshproxy

```
abe$ sshproxy.sh
Enter your password+OTP: NIM.password157712
Successfully obtained ssh key /Users/abe/.ssh/nersc
Key is valid: from 2019-01-23T04:36:00 to 2019-01-24T04:37:51
abe$ ls ~/.ssh
config     id_rsa.pub nersc      nersc.pub
id_rsa     known_hosts nersc-cert.pub
 abe$ ssh -i ~/.ssh/nersc cori.nersc.gov
 *****************************************************************
 *                                                               *
 *                      NOTICE TO USERS                          *

 abe@cori07:~>
```

# Less typing...

In ~/.ssh/config

```
Host cori cori.nersc.gov
        Hostname cori.nersc.gov
        IdentityFile ~/.ssh/nersc
        AddKeysToAgent no
```

```
abe$ ssh cori
*****************************************************************
*                                                               *
*                      NOTICE TO USERS                          *


abe@cori07:~>
```

# SSHProxy and ssh-agent

Adding sshproxy keys to ssh-agent can have undesirable results

- ssh tries *every key* in ssh-agent until one matches
- most ssh servers have **`MaxAuthTries 6`**
  - And then you just get a generic authentication failure
- ssh-agent doesn't respect certificate expiration
  - expired keys get tried and fail

You probably don't need to use ssh-agent

- sshproxy private keys are unencrypted
- Hostbased within NERSC
- but if you really need to: **`sshproxy.sh -a`**

# Options to sshproxy.sh

```
Usage: sshproxy.sh [-avh][-u <user>][-o <filename>][-s <scope>][-U <server
URL>]

     -u <user>        Specify remote (NERSC) username (default: abe)
     -o <filename>    Specify pathname for private key (default:
                      /Users/abe/.ssh/nersc)
     -s <scope>       Specify scope (default: 'default')
     -a               Add key to ssh-agent (with expiration)
     -v               Print out version number and exit
     -U <URL>         Specify alternate URL for sshproxy server (generally
                      only used for testing purposes)
```

# Scopes

sshproxy can provide keys with longer expiration times

- for longer running jobs and other automated workflows

We create a *scope* for you

- A scope is a name that you provide to sshproxy.sh
- The scope determines how long the key is valid for (e.g. one week)
- Each scope has a set of users authorized to use it
- You can request a key with a given scope as often as you wish
- Scope *permissions* expire (typically one year)

```
sshproxy -s myscope -o ~/.ssh/one-week-key
```

# Common Problems

- Username on your workstation doesn't match NERSC username
  - `sshproxy.sh -u abe`

- ssh still prompts for password+OTP
  - Did you use the "-i" flag? `ssh -i ~/.ssh/nersc`
  - Or see instructions for modifying .ssh/config

- Getting "authentication failed"
  - Too many keys in ssh-agent? `ssh-add -L`
  - Purge keys: `ssh-add` -D
  - Remove `AddKeysToAgent yes` from `.ssh/config`

NERSC

# Thank You