CGI

Experience the commitment®

Rehabilitating Pickle

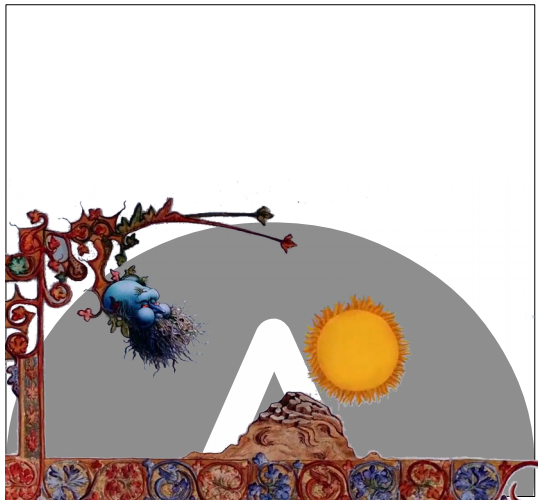
Alex Willmer
Developer, CGI
PyConUK 2018

Previously at CGI

DevOps with Ansible

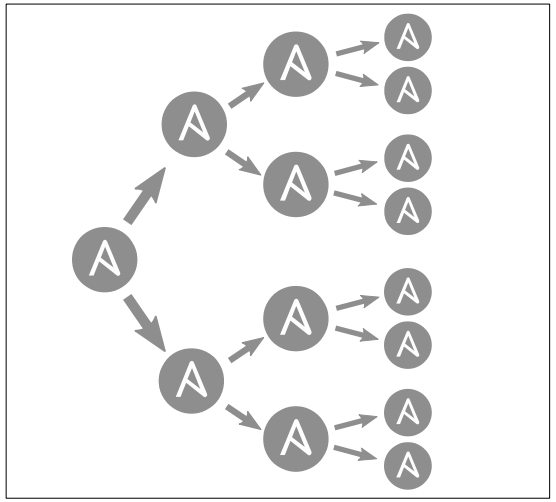
There is much rejoicing

(Yayyy)

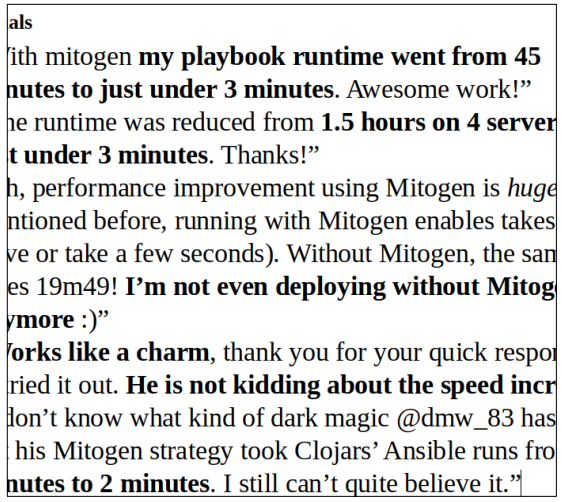


Playbooks grow, get slow

Summer becomes Winter



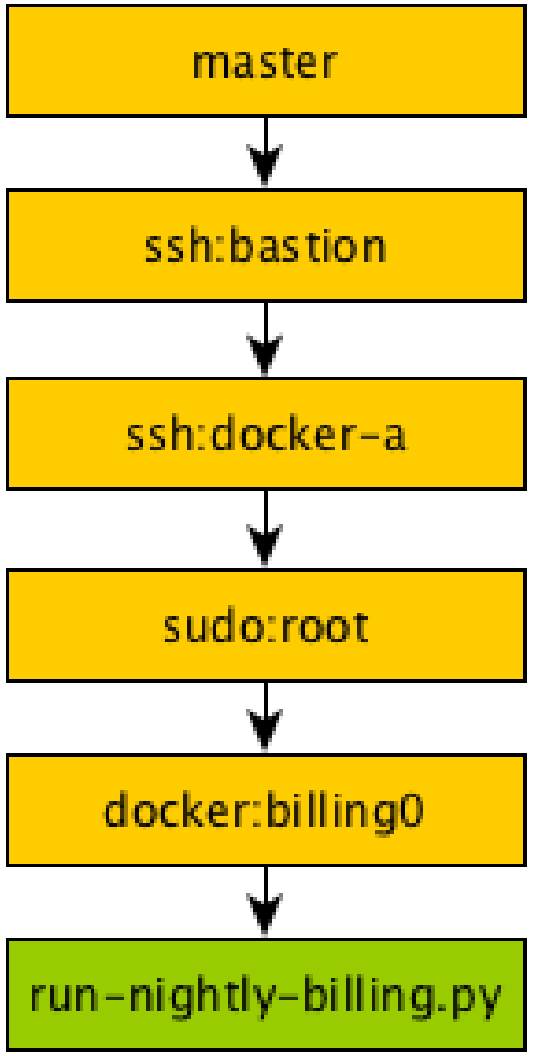
Mitogen for Ansible appears



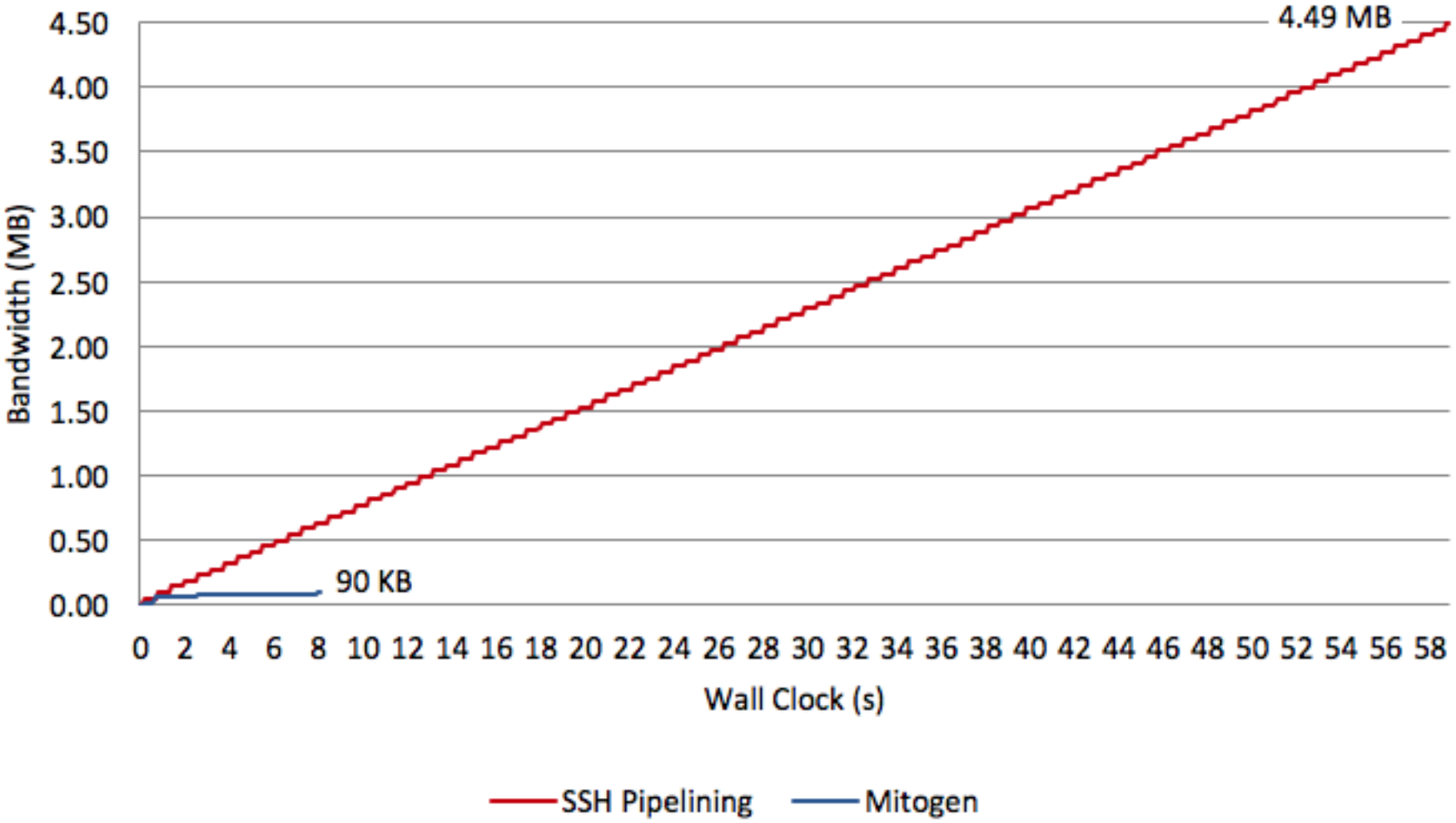
Ansible is fast again …

Mitogen

```
bastion = router.ssh(
        hostname='jump-box.mycorp.com')

docker_host = router.ssh(
        via=bastion_host,
        hostname='docker-a.prod.mycorp.com')

sudo_account = router.sudo(
        via=docker_host,
        username='azure_diamond', password='hunter2')

internal_box = router.docker(
        via=sudo_account,
        container='billing0')

internal_box.call(os.system, './run-nightly-billing.py')
```



Mitogen makes Ansible faster



Pickle

```
>>> original = (1.0, 'text', [2,], {3: None})

>>> pickle.dumps(original, protocol=0)
b'(F1.0\nVtext\np0\n(lp1\nL2L\na(dp2\nL3L\nNstp3\n.'

>>> pickle.loads(_)
(1.0, 'text', [2,], {3: None})
```

Why Pickle?

- Preserves nearly any type or class
  - JSON can’t do tuple(), set(), complex(), large integers, some dict
- Preserves self referencing data structures
  - JSON doesn’t handle recursive structures
- Works everywhere Python does
- Produces compact serializations

DON'T USE PICKLE

Why not Pickle?

> ***Warning:*** *The pickle module is not secure against erroneous or maliciously constructed data. Never unpickle data received from an untrusted or unauthenticated source.*
>
> Python documentation » 12.1 pickle — Python object serialization

Why not Pickle?

> ***Warning:*** *The pickle module is not secure against erroneous or maliciously constructed data. Never unpickle data received from an untrusted or unauthenticated source.*
>
> Python documentation » 12.1 pickle — Python object serialization

CVE-2005-2875	CVE-2005-3008	CVE-2007-1100	CVE-2007-5741	CVE-2008-3143	CVE-2010-4574
CVE-2011-1113	CVE-2011-2520	CVE-2012-4406	CVE-2013-5093	CVE-2013-5942	CVE-2014-0485
CVE-2014-1714	CVE-2014-3539	CVE-2014-5340	CVE-2014-8165	CVE-2015-0692	CVE-2015-0693
CVE-2015-5164	CVE-2015-5242	CVE-2016-3957	CVE-2017-10803	CVE-2018-7889	CVE-2018-14572

– US National Vulnerabilities Database (NVD), vulnerabilities involving "pickle"

pickle.load() executes arbitrary code

```
>>> pickle.loads(b"cos\nsystem\n(S'rm -rf'\ntR.")
0
```

Demo

Restricting globals

```
from pickle import Unpickler, UnpicklingError

class RestrictedUnpickler(Unpickler):
    def find_class(self, module, name):
        if module != "builtins":
            raise UnpicklingError("Denied!")
        elif name not in {"set", "frozenset"}:
            raise UnpicklingError("Still no")
        return getattr(builtins, name)
```

Other attacks

Denial of service

- Protocol downgrade
- Billion Laughs (pickle bombs)
- Unhandled exceptions

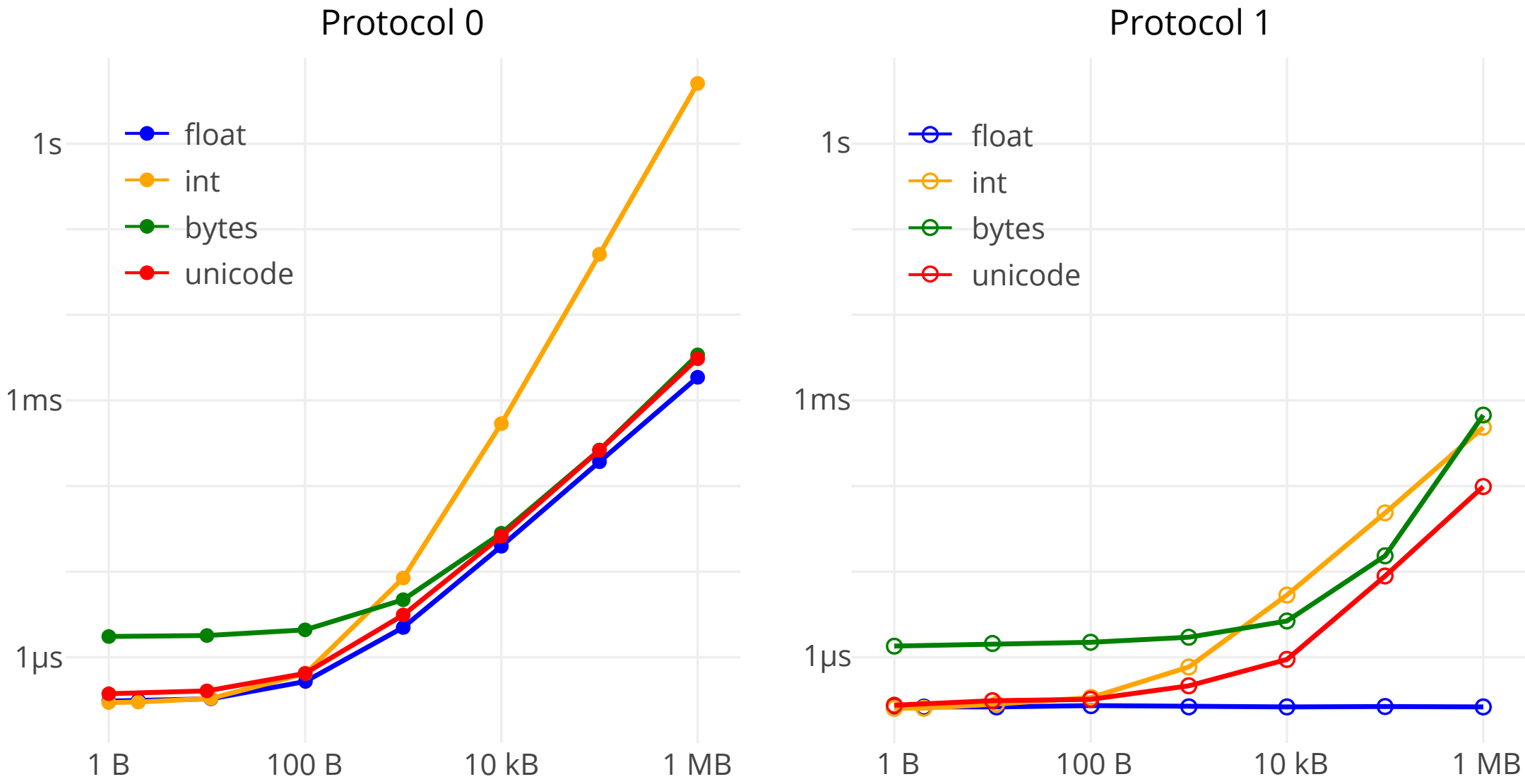
Data Exposure

- Object traversal (gadget chains)

Weird machines

- Unused opcodes (e.g. DUP)
- Parser abuse
- Stack corruption

Protocol downgrade

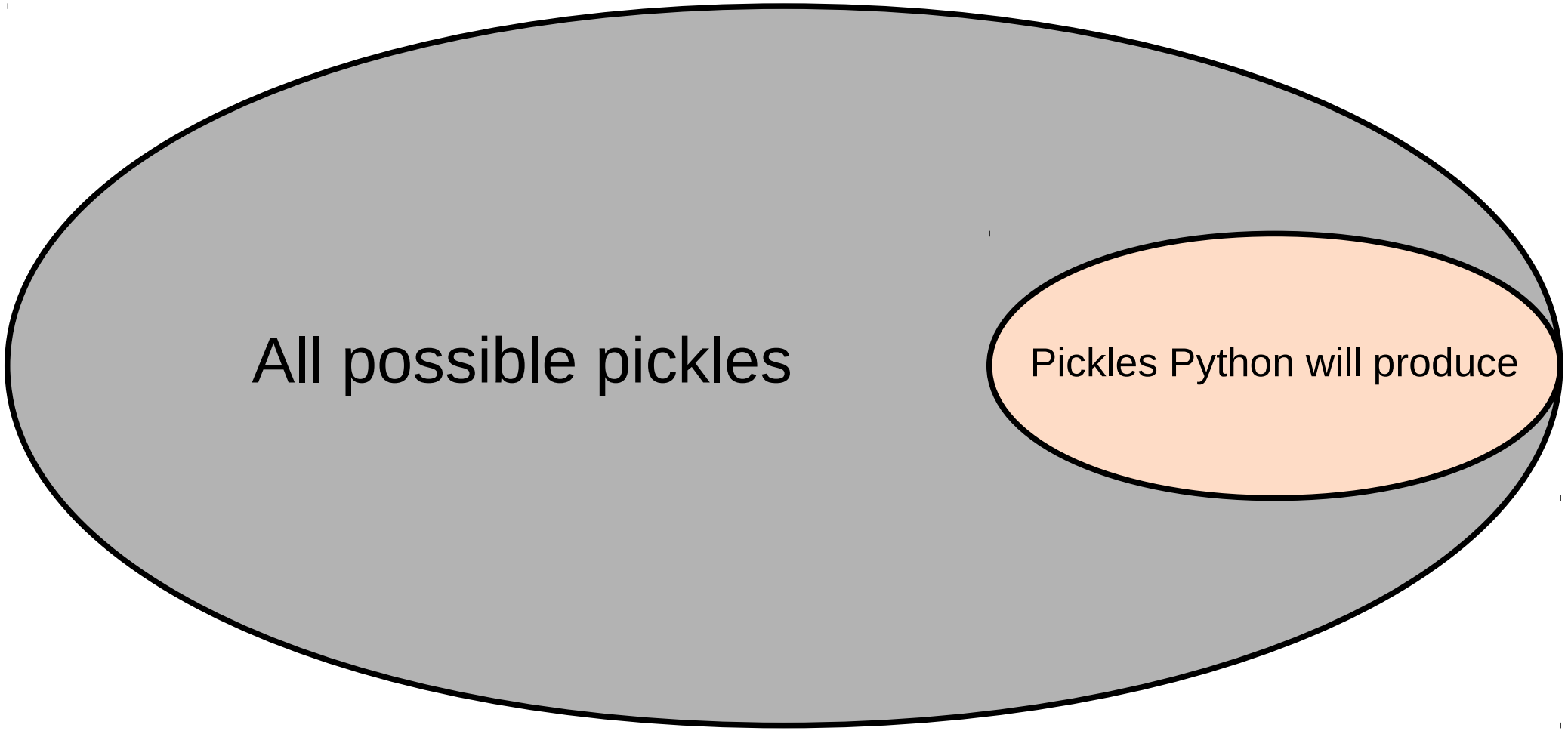


Billion laughs / pickle bombs

```
>>> a = ['lol', 'lol', 'lol', 'lol', 'lol',
...      'lol', 'lol', 'lol', 'lol', 'lol']
>>> b = [a,a,a,a,a,a,a,a,a,a]
>>> c = [b,b,b,b,b,b,b,b,b,b]
...
>>> laughs = [i,i,i,i,i,i,i,i,i,i]
>>> with open('billion-laughs.pkl0', 'wb') as f:
...     pickle.dump(laughs, f)
```

Demo

Weird machines



Other considerations

Don’t use pickle for long term storage

> *Its automatic, magical behavior shackles you to the internals of your classes in non-obvious ways. You can’t even easily tell which classes are baked forever into your pickles. Once a pickle breaks, figuring out why and where and how to fix it is an utter nightmare.*
>
> – Don’t use Pickle – use Camel, Eevee

Thank you

Slides, etc.

- github.com/moreati/pickle-fuzz

Mitogen (by David Wilson)

- mitogen.readthedocs.io

Pikara (by Latacora)

- github.com/latacora/pikara

Sour pickles, real world pickle attacks (by Sensepost)

- sensepost.com/blog/2011/blackhat-2011-presentation/
- youtube.com/watch?v=HsZWFMKsM08

Don’t use pickle – use camel (by Eevee)

- eev.ee/release/2015/10/15/dont-use-pickle-use-camel/

CGI are hiring

Alex Willmer
Developer, CGI
alex.willmer@cgi.com
@moreati