HTCondor and Networking

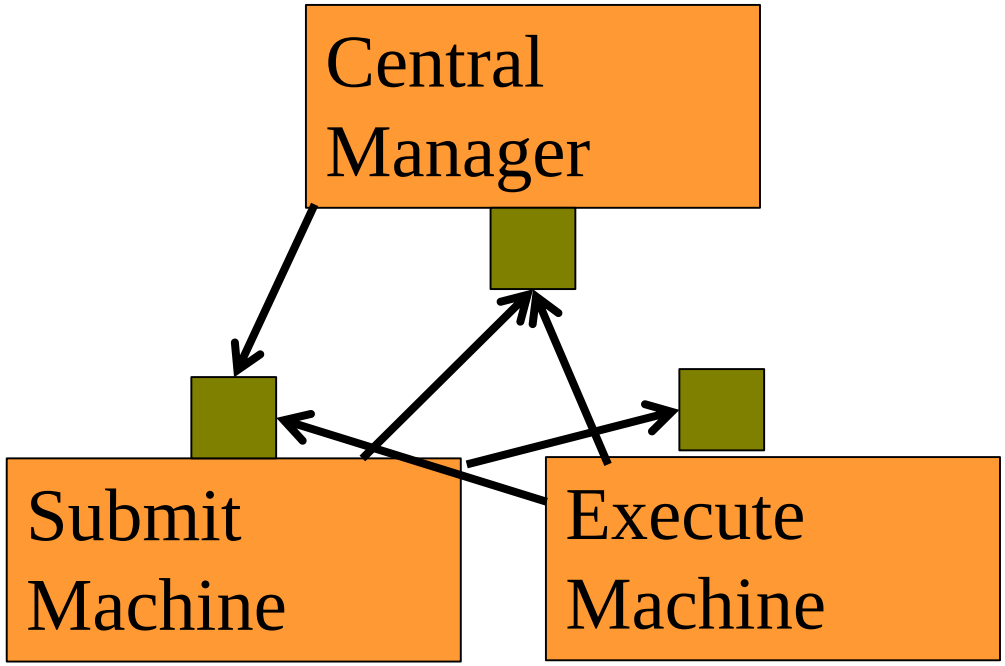
Jaime Frey
Center for High Throughput Computing

Introduction

- HTCondor built in a simpler time:
  - Every machine can connect to every other
  - More TCP ports available than can be used
  - Every machine has 1 network interface
  - IPv4 “enough addresses for everyone”
  - DNS exists everywhere, correctly and reliably
  - All connections symmetric

Design Problem: Listeners everywhere

- Multihoming?
- Firewalls?
- NAT?
- Asymmetry?



Each daemon has ONE address in collector! (mostly)

What is “the name”?

The “sinful” string:

examples

<192.168.1.15:9618>

<192.168.1.15:9618?key=value>

In MyAddress attribute

And condor_tool –addr ‘<sinful>’

Which Address will a machine advertise?

If…

`BIND_ALL_INTERFACES = true` (default)

`NETWORK_INTERFACE = unset` (default)

`ENABLE_ADDRESS_REWRITING = true` (default)

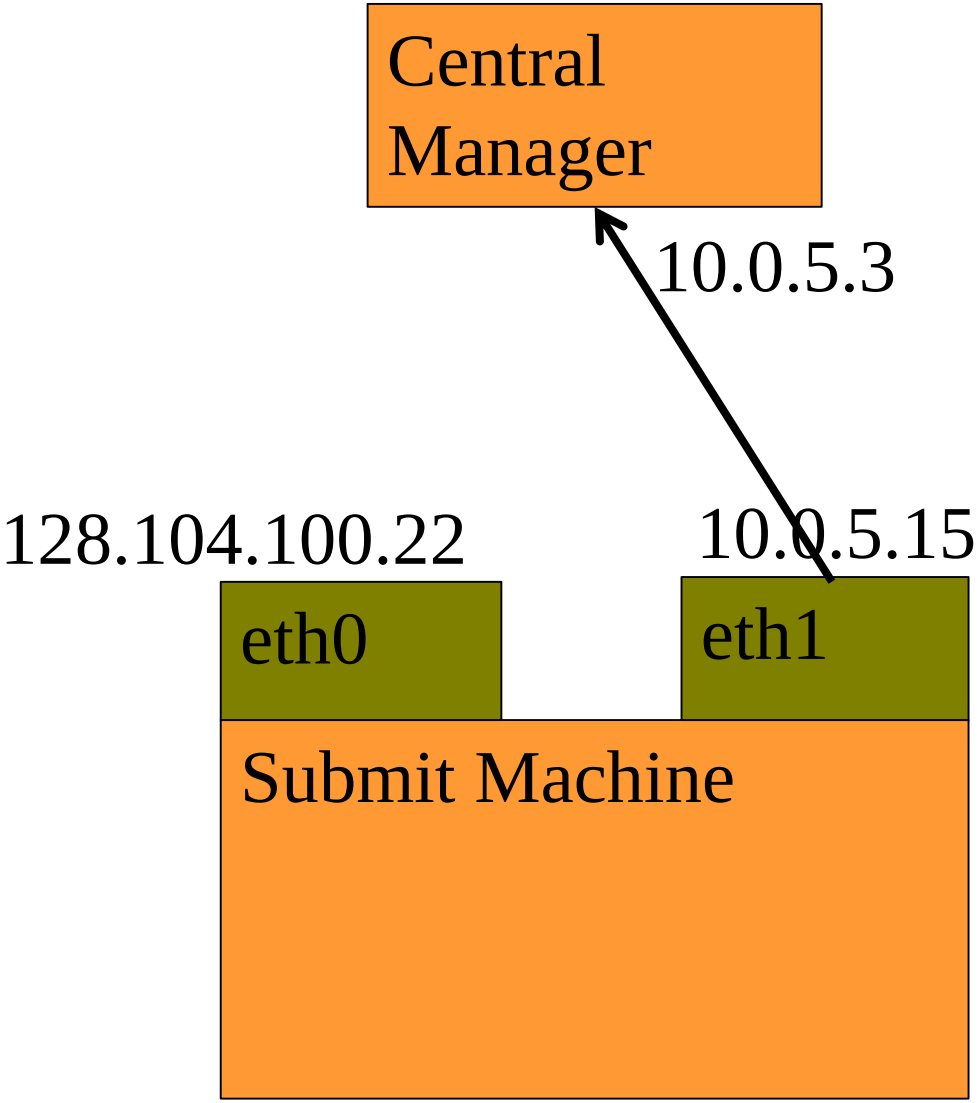
Then…

Machine **listens** on all interfaces,

Prefers most “public” interface locally,

Uses “collector” interface when advertising

Network rewrite



Which Address will a machine advertise?

If…

```
BIND_ALL_INTERFACES = false
```
(undefault)

```
NETWORK_INTERFACE = 10.*
```
(or)

```
NETWORK_INTERFACE = eth0
```
(or)

```
NETWORK_INTERFACE = 10.5.3.4
```

Then…

Machine **listens** on specified interface (only), and advertises that!

Which Address will a machine advertise?

If…

`BIND_ALL_INTERFACES = false` (undefault)

`NETWORK_INTERFACE =` <unset> (default)

Then…

Machine **listens** on one interface (the most "public" one) and advertises that.

Completely Punting to proxy

- `TCP_FORWARDING_HOST = foo.com`

- Says “you can connect to me at foo.com”
  - IP address of foo.com is advertised
- How?
  - Up to you:
    - Ssh forwarding
    - iptables?
    - EC2 public address

Solutions for firewalls

- Easiest: HIGHPORT/LOWPORT
  - `LOWPORT = 9000`
  - `HIGHPORT = 10000`
- Assuming holes punched in firewall
- If only need inbound (common case):
  - `IN_LOWPORT = 9000`
  - `IN_HIGHPORT = 10000`

How Many ports?

- Schedd:
  - 5 + 2 * MAX_JOBS_RUNNING
- Startd
  - 5 + 2 * max slots
- (Assuming no shared_port or CCB)

What happens on port exhaustion?

- Badness.
- Jobs will fail to start for no apparent reason

- Keep an eye on ports in this case.

Private network support

```
PRIVATE_NETWORK_INTERFACE = 1.2.3.4
PRIVATE_NETWORK_INTERFACE = eth1
PRIVATE_NETWORK_NAME = MyPrivNet
```

If two machines have the same private network name, they will use the private address to communicate.

Need not actually be a *private* network

Shared Port

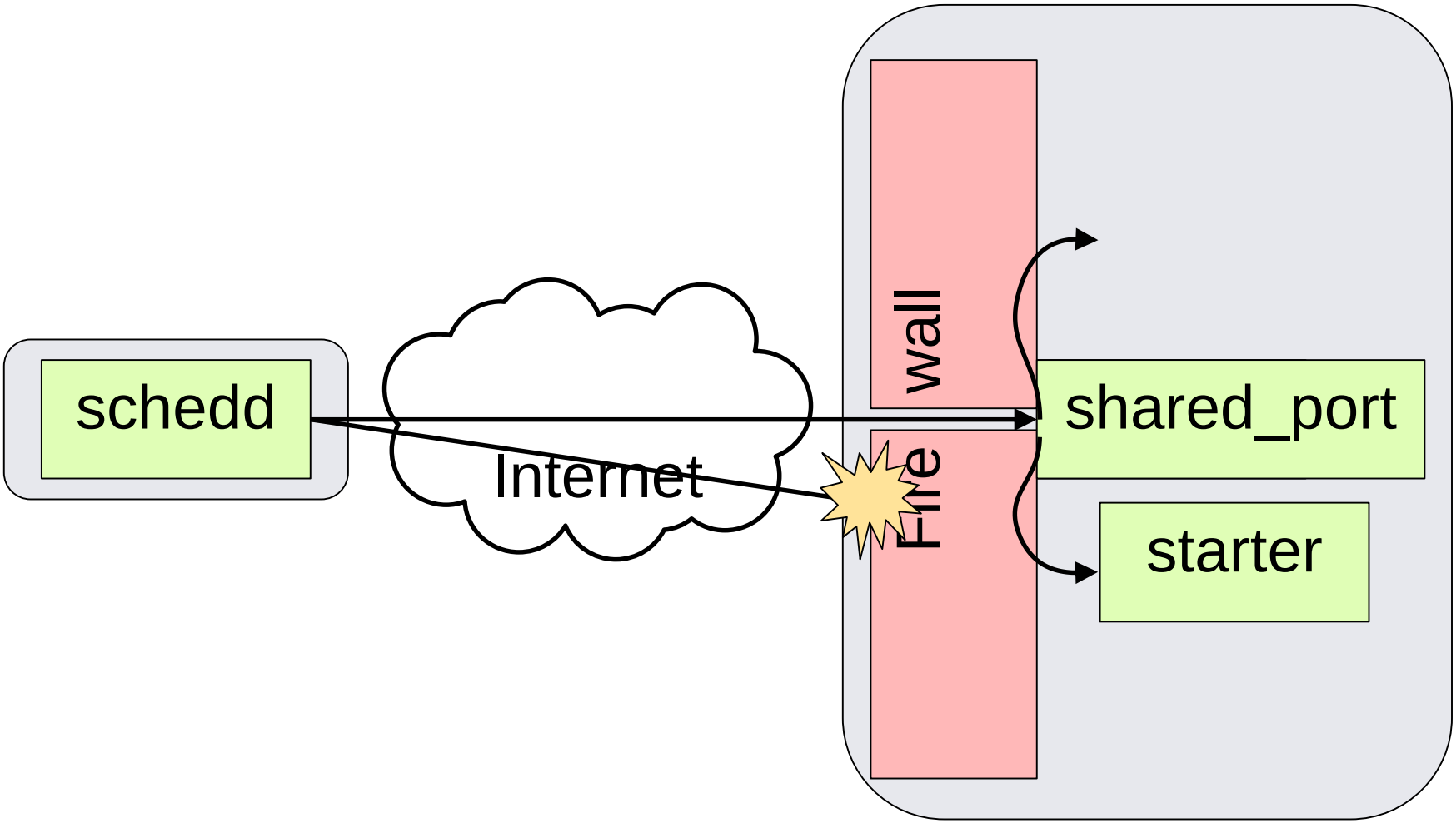
- Problem: only ~ 60,000 TCP ports
- Need one per shadow
- Shared port Service
  - *Doesn't work with standard universe*

`USE_SHARED_PORT = true` (default in 8.5.1)

- Open single port in firewall
- Changes sinful string to

`<192.168.1.100:9618?sock=xxx_yyy>`

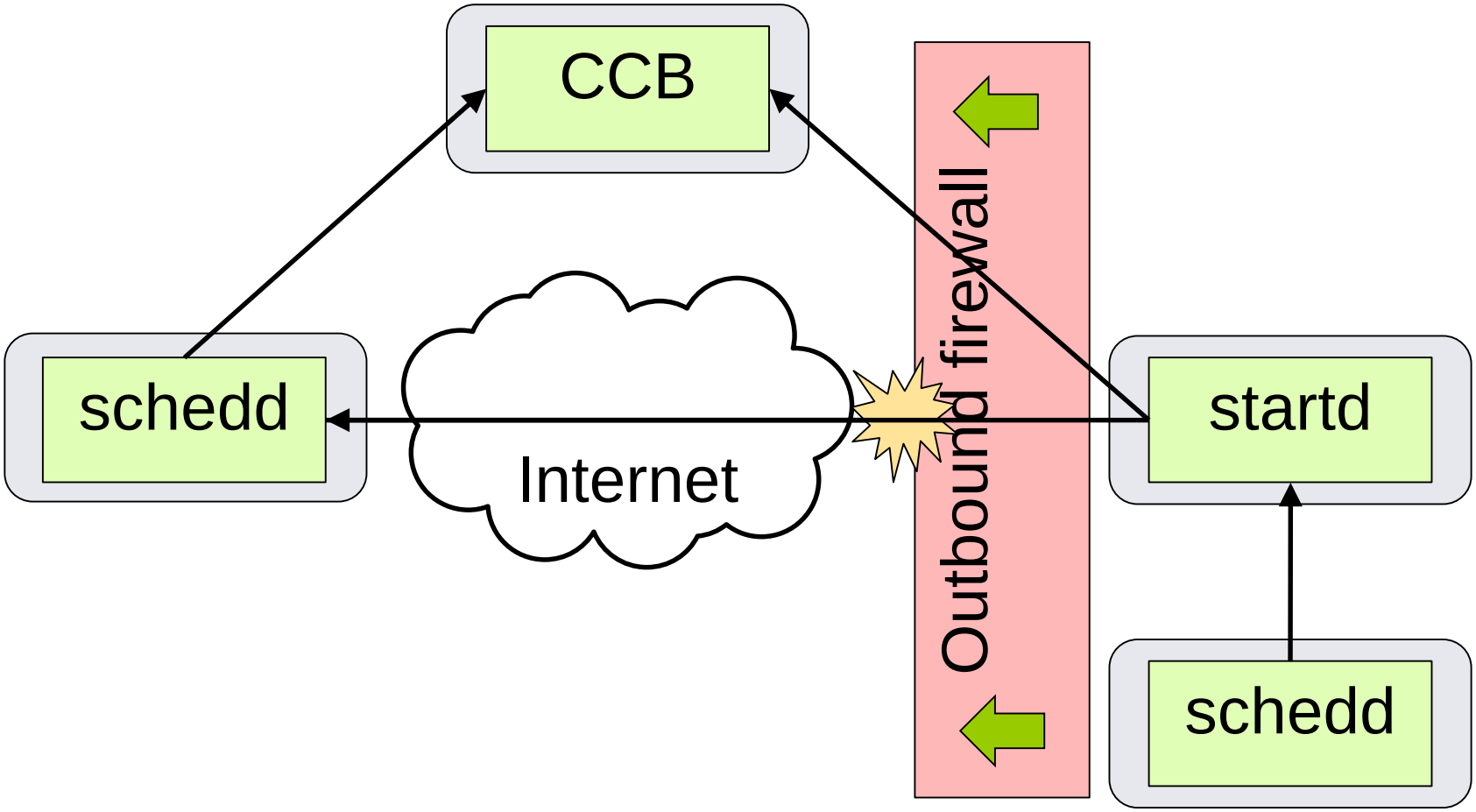
condor_shared_port



CCB: Condor Connection Broker

- Bypasses firewalls by reversing connection
- Requires one machine with no firewall
  - Usually the collector
- Doesn't work with standard universe
- Only bypasses one firewall
  - Usually in front of the startds
  - Schedds / Central managers w/o firewalls (or firewall with single hole for shared port)

CCB: Condor Connection Broker



CCB Configuration

- CCB built into condor_collector

```
CCB_ADDRESS = $(COLLECTOR_HOST)
PRIVATE_NETWORK_NAME = domain
```

- Machine behind same firewall can communicate directly

IPv6

- IPv6-only mode
  - `ENABLE_IPV6 = true`
  - `ENABLE_IPV4 = false`
- Network parameters work as before
  - `NETWORK_INTERFACE = 2607:f388:1086:0:21b:24ff:fedf:b520`

IPv4/IPv6 Mixed Mode

`ENABLE_IPV4 = True` (default)

`ENABLE_IPV6 = True` (default in 8.5.3)

- Both interfaces advertised, IPv6 preferred
- Central managers and submit machines must support both
- Execute machines can be IPv4-only or IPv6-only
- Ease transition to IPv6
  - `PREFER_IPV4 = true`

Putting it all together

- CCB works with shared port
  - Common Combination
- If you have CCB or shared port, probably don't need highport/lowport
- CCB works together with private networks
  - Can be big performance win

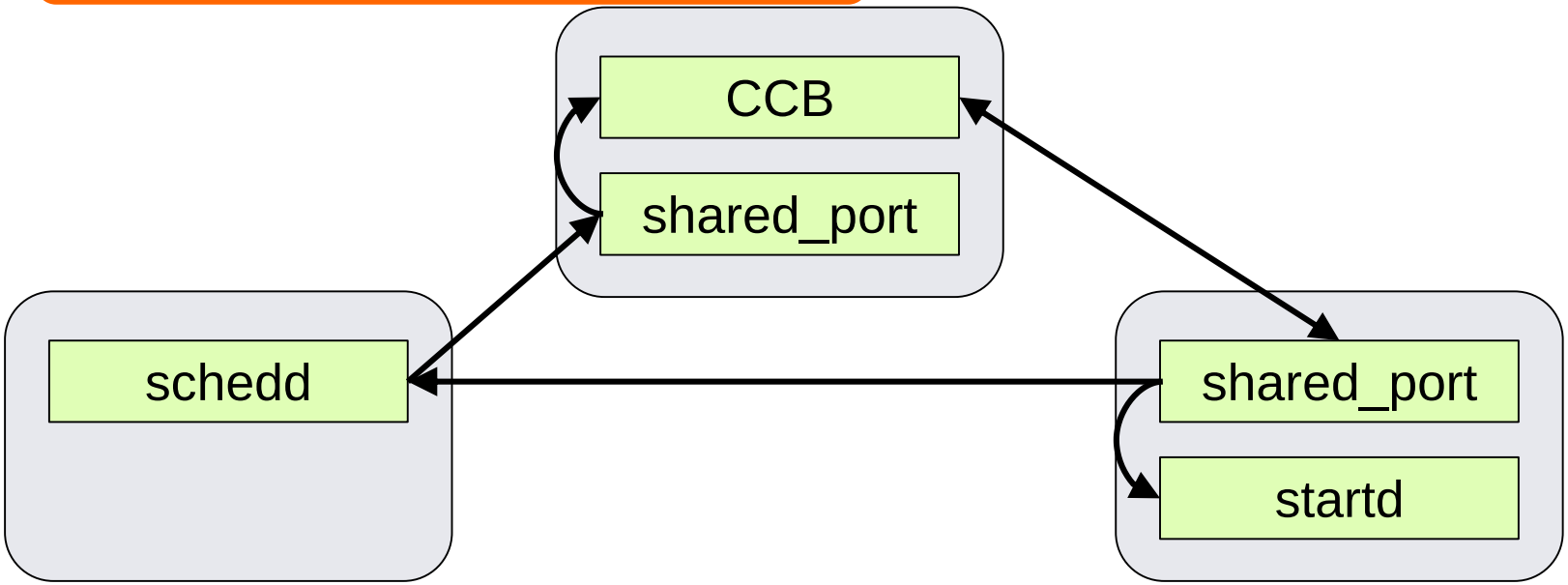
Multi-Stage Routing

<192.168.1.55:9618?

CCBID=173.194.46.96:80%23**381**%3F

sock%3D917_aa8b_3%26

sock=1567_808b_3>



Thank you!