



HTCondor and Networking

Greg Thain

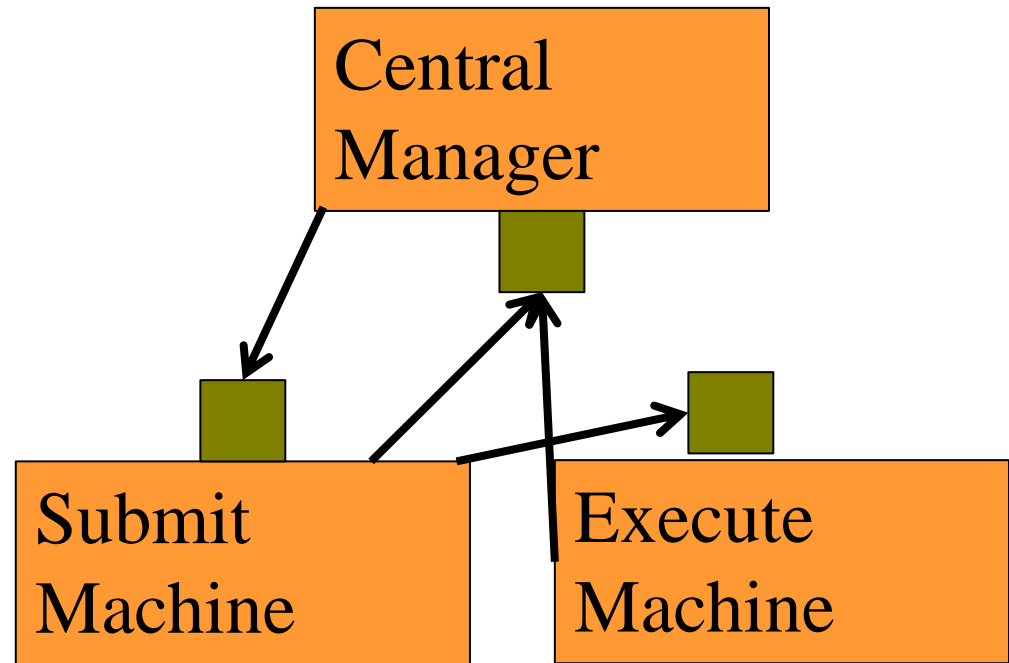
Center for High Throughput Computing

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!

What is “the name?”

The “sinful” string:

examples

<192.168.1.15:9618>

<192.168.1.15:9618?key=value>

In MyAddr attribute

And condor_tool –addr ‘<sinful>’

Which Address will a machine advertise?

If...

`BIND_ALL_INTERFACES = true` (default)

`NETWORK_INTERFACE = unset` (default)

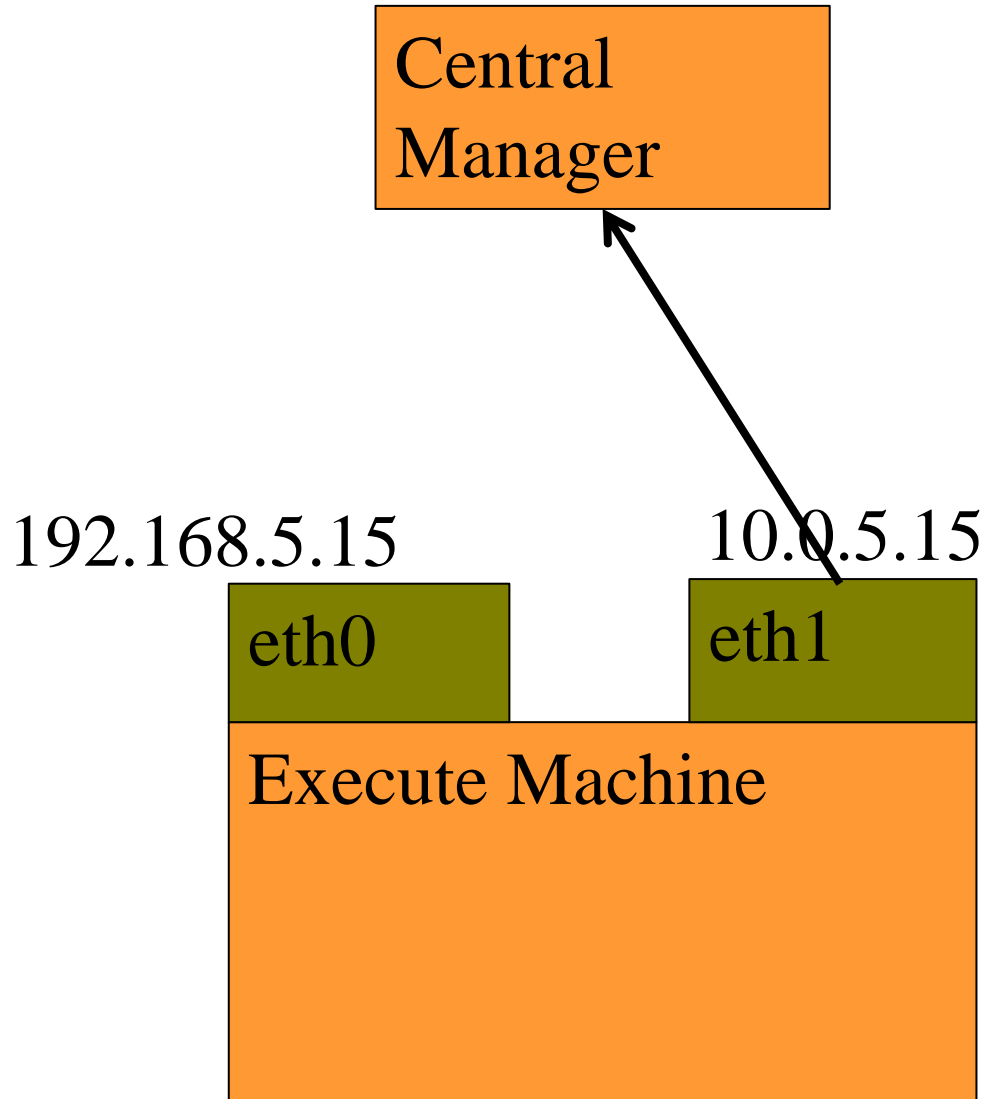
`ENABLE_ADDRESS_REWRITING = true` (default)

Then...

Machine **listens** on all interfaces,

Collector rewrites to “collector” interface

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(default)
```

```
NETWORK_INTERFACE = unset (default)
```

Then...

Machine **listens** on one interface, heuristically chosen by condor, and advertises that.

Completely Punting to proxy

- › `TCP_FORWARDING_HOST = foo.com`
- › Says “you can connect to me at foo.com”
- › How?
 - Up to you:
 - Ssh forwarding
 - iptables?
 - Magic

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

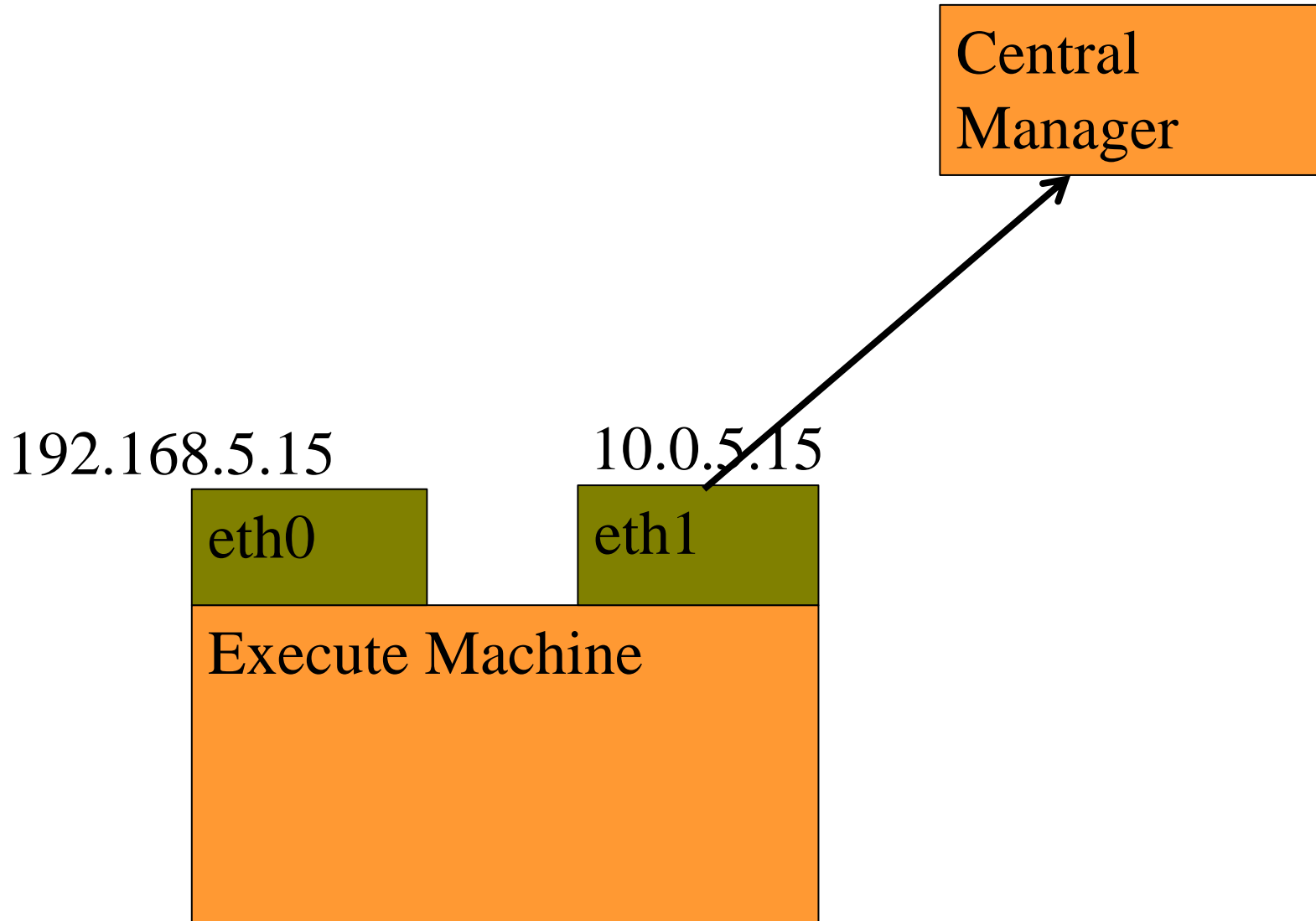
How Many ports?

- › Schedd:
 - $5 + 5 * \text{MAX_JOBS_RUNNING}$
- › Startd
 - $5 + 5 * \text{max slots}$
- › (Assuming no shared_port or CCB)

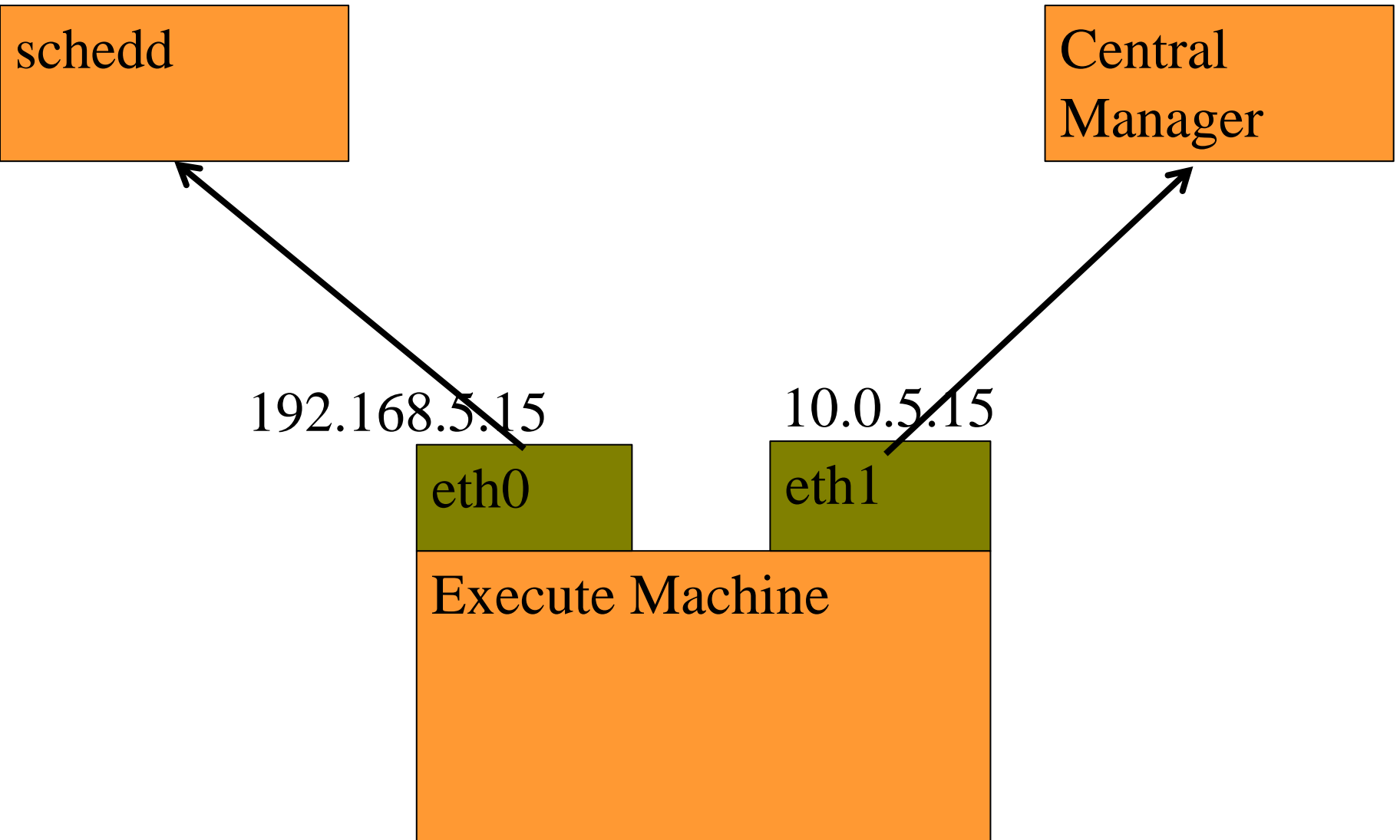
What happens on port exhaustion?

- › Badness.
- › Jobs won't start for no apparent reason
- › Keep an eye on ports in this case.

Split Network



Split Network



Private network support

```
PRIVATE_NETWORK_INTERFACE =  
1.2.3.4
```

```
PRIVATE_NETWORK_INTERFACE=eth1
```

```
PRIVATE_NETWORK_NAME=MyPrivNet
```

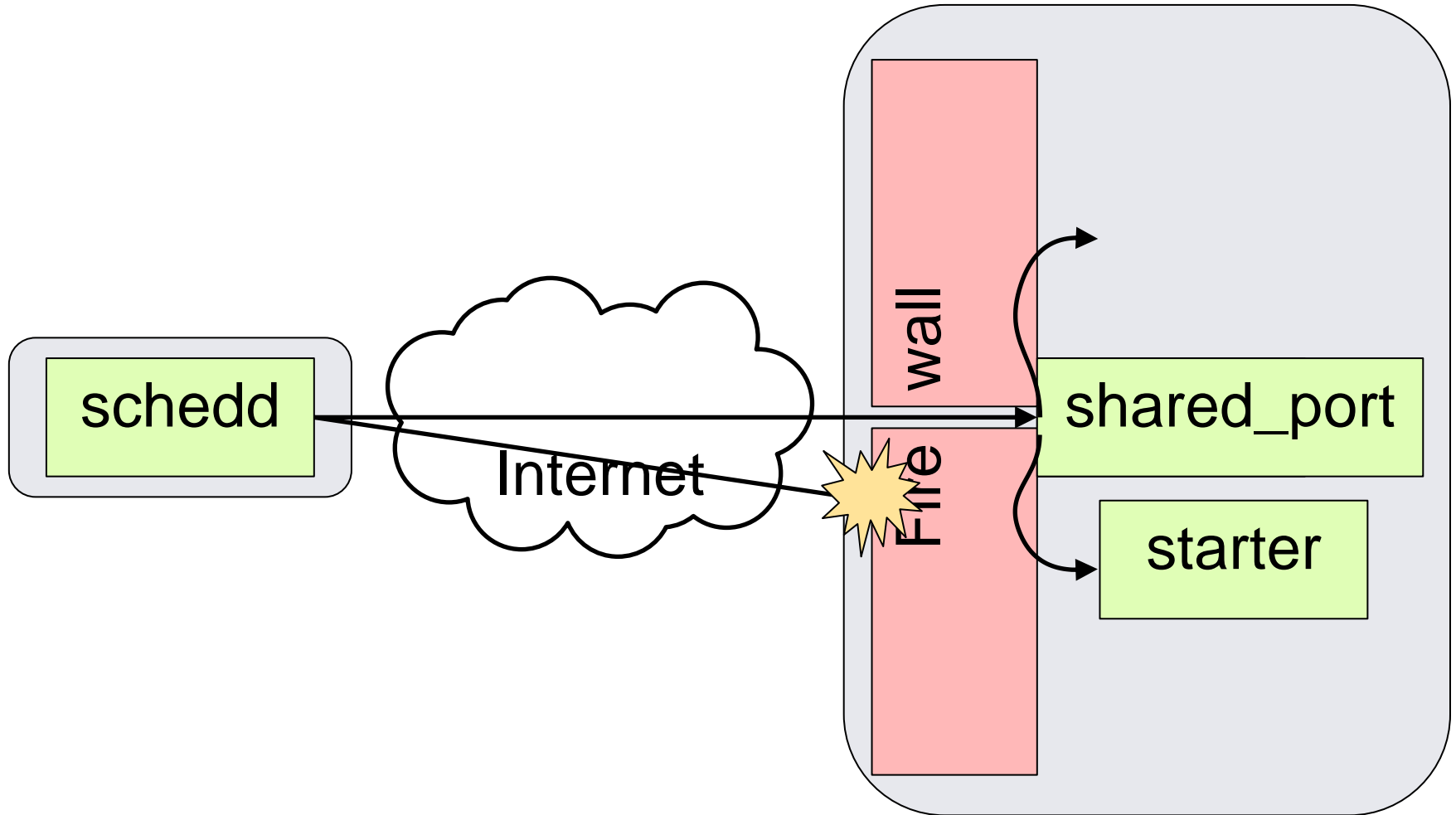
Any time two condor machine connect,
condor will use this network and advertise it.

Need not actually be the *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`
- › `DAEMON_LIST = ... SHARED_PORT`
- › Changes sinful string to
`<192.168.1.100:9618?sock=xxx_yyy>`

condor_shared_port

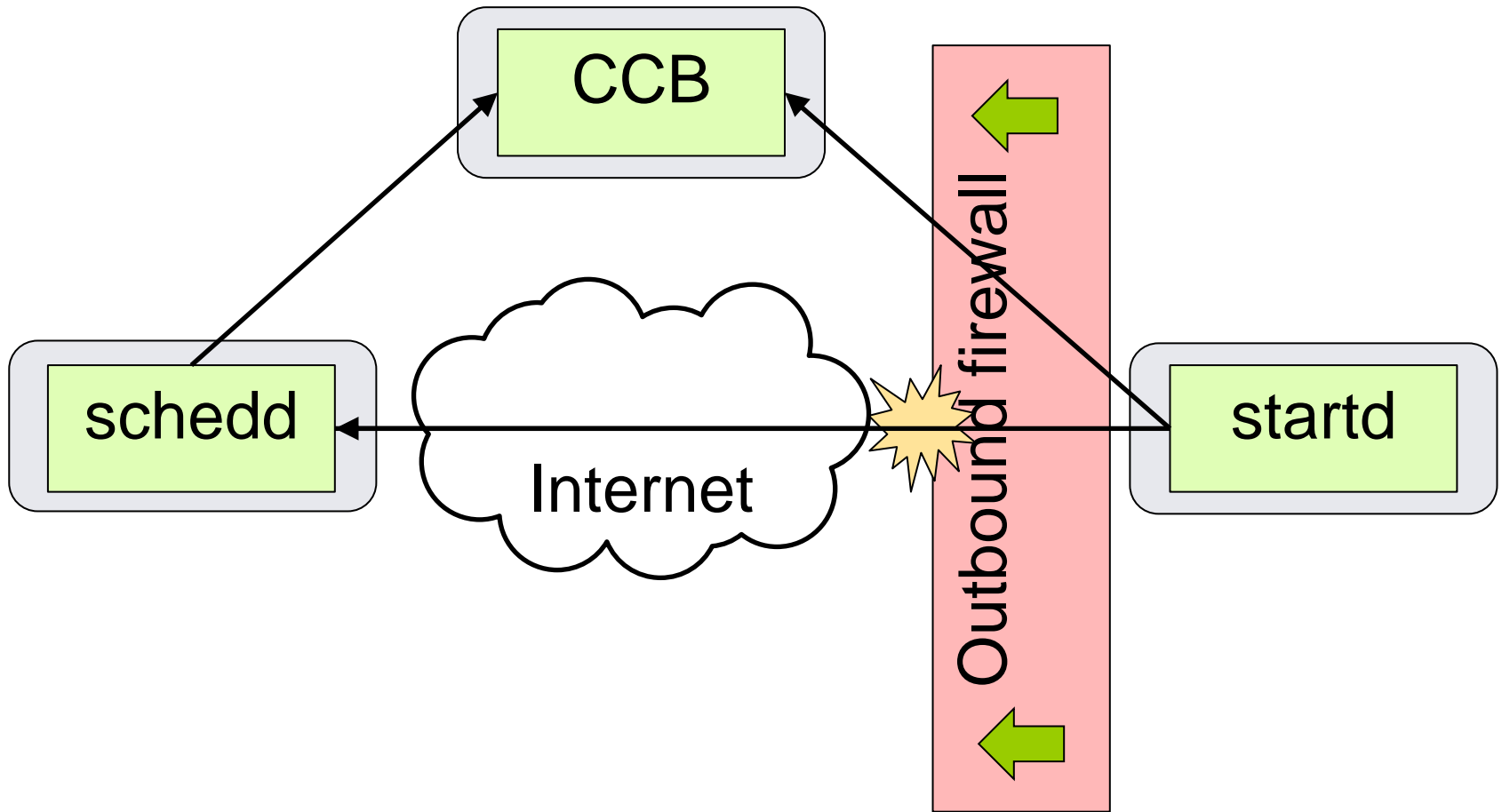


CCB:

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

CCB: Condor Connection Broker



CCB Configuration

- › CCB built into condor_collector

```
CCB_ADDRESS = $(COLLECTOR_HOST)
```

```
PRIVATE_NETWORK_NAME = domain
```

IPv6

- › Still an active area of work

```
ENABLE_IPV6 = true
```

```
ENABLE_IPV4 = false
```

```
NETWORK_INTERFACE = \  
2607:f388:1086:0:21b:24ff:fedf:b520
```

Putting it all together

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

Thank you!