

NERVOUS SYSTEM

Nerves Clustering Using Your Own Distribution

by

Timmo Verlaan (@diptimmo)

HOW IT STARTED

HELLO BLINKY

BREWING ELIXIR

HELLO NETWORK

NERVES WIFI

**HELLO
DISTRIBUTION?**

DISTRIBUTED BLINKY



config.exs

```
host_id = Mix.Project.config[:host_id]

# generate 'assumed' initial node list
host_id_int = String.to_integer(host_id)
initial_nodes = for id <- 2..host_id_int do
  String.to_atom("#{id}@10.0.10.#{id}")
end

# configure this node
config :distributed_blinky, if_eth0: [
  ip: "10.0.10.#{host_id}",
  ...
],
initial_nodes: initial_nodes
```

vm.args

```
## Start the Elixir shell
-name <%= host_id %>@10.0.10.<%= host_id %>
-cookie distributed_blinky
-noshell
-user Elixir.IEx.CLI
-extra --no-halt
```

NAMING

- -name
- -sname

EPMD

Erlang Port Mapper Daemon

4369

INET_TCP_DIST

CUSTOM DISTRIBUTION

-PROTO_DIST

<NAME_OF_MODULE>_dist

module_dist.ex

```
# callback for setting up new connection
def setup(node, type, my_node, :shortnames, setup_time) do

  # get epmd module
  mod = :net_kernel.epmd_module()

  # ask epmd module for address and port
  {ip, port} = mod.address_and_port_please(node)

  ...
end
```


-EPMD_MODULE

epmd.ex

```
def address_and_port_please(name) do
  GenServer.call(__MODULE__, {:address_and_port_please, name})
end

def handle_call({:address_and_port_please, name}, _f, state) do
  {ip, port} = extract_ip_port(state)
  {:reply, {ip, port}, state}
end
```

THANK YOU!

- @diptimmo
- github.com/tverlaan/inet_tcp_dist
- tverlaan@gmail.com