

mini-ndn - Task #4078

Use popen in ndn_application

05/05/2017 01:30 PM - Ashlesh Gawande

Status: New	Start date: 05/05/2017
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 0.00 hour
Target version: v0.7.0	

Description

Currently we use cmd. Then we use a shell command to get the pid - which is unreliable.

But Popen needs to be run with correct HOME environment, otherwise NLSR can't connect with NFD.

Popen does not seem to work correctly for cluster, even when HOME environment is passed:

```
out = self.node.popen("printenv").stdout.read()
env = {}
for elem in out.split("\n"):
    if elem != "":
        tmp=elem.split("=")
        env[tmp[0]] = tmp[1]
        env["HOME"] = self.node.homeFolder
self.popen = self.node.popen(command.split(), env=env)
```

So maybe for now keep using cmd for cluster:
https:// Gerrit: named-data.net/#/c/3870/3/ndn/ndn_application.py

Related issues:

Related to mini-ndn - Task #3051: Investigate Mininet's per-host private dire... Code review 07/17/2015

History

#1 - 05/24/2017 06:34 AM - Junxiao Shi

[mnndn](#) has used popen for years.

The trick is [passing privateDirs to mininet.node.Host constructor](#), so that \$HOME can keep its default value of /root and does not need to be overridden.

#2 - 05/24/2017 08:21 AM - Ashlesh Gawande

- Related to Task #3051: Investigate Mininet's per-host private directories feature for NFD socket file added

#3 - 12/21/2017 04:06 PM - Ashlesh Gawande

- Target version set to v0.7.0

#4 - 06/07/2018 07:40 AM - Junxiao Shi

Popen does not seem to work correctly for cluster, even when HOME environment is passed:

Mininet Cluster Edition implements [RemoteMixin.popen](#) as follows:

```
def _popen( self, cmd, sudo=True, tt=True, **params):
    if self.isRemote:
        if sudo:
            cmd = [ 'sudo', '-E' ] + cmd
        if tt:
            cmd = self.sshcmd + cmd
        else:
            # Hack: remove -tt
            sshcmd = list( self.sshcmd )
            sshcmd.remove( '-tt' )
            cmd = sshcmd + cmd
```

```
else:
    if self.user and not sudo:
        # Drop privileges
        cmd = [ 'sudo', '-E', '-u', self.user ] + cmd
    popen = super( RemoteMixin, self )._popen( cmd, **params )
    return popen
```

When you pass `env` to `node.popen`, the environment variables apply to "ssh" process, but the remote command (such as `/usr/bin/nfd`) does not receive them.

You could set environment variables in the command line before passing them to `node.popen`, but it's better to patch `RemoteMixin._popen` with this feature and send a pull request.