General scripts for CLTL

Paul Huygen

4th February 2018

Abstract

this document describes and generates scripts that are useful.

1 Pipeline

1.1 Restore the eSRL server

It turns out that the server for 'eSRL' module may stop to work as advertised. An example of an error message that is caused by a corrupted 'eSRL' server:

```
java.net.SocketException: Connection reset
        at java.net.SocketInputStream.read(SocketInputStream.java:196)
        at java.net.SocketInputStream.read(SocketInputStream.java:122)
        at java.net.SocketInputStream.read(SocketInputStream.java:210)
        at java.io.DataInputStream.readBoolean(DataInputStream.java:242)
        at ixa.srl.SRLClient.main(SRLClient.java:70)
Completed: module /usr/local/share/pipelines/nlpp/bin/eSRL; result 0
```

The problem can be solved by shutting down the `eSRL` server. To this end there is a script `kill_eSRL_server` that does this, provided the issuer belongs to the "sudo" group.

This script does the following: First it finds out which process listens on the port on which `eSRL` listens, i.e. port 5005. The it checks whether the command-line instruction for that process is indeed command to start the eSRL server. If that is the case, the script kills the process. The Next time that `nlpp` runs it will automatically start a new `eSRL` server.

To find out whether a process listen on port 5005 and, if so, what the ID of that process is, the script issues the `netstat` command. This command produces a line that looks like

```
tcp 0 0 0.0.0.0:5005 0.0.0.0:* LISTEN 29891/java |
```

The process-id of the listener is located before the string `/java`. The following `awk` script extracts that number:

⟨ *define awk-script to extract eSRL process-id* 1 ⟩ ≡
```
    awkscript='{match($7, /([[:digit:]]+)\/[.]*/, arr); print arr[1]}'
```
    ⋄

Fragment referenced in 2.

The command that starts the `eSRL` server has form like:

```
java -Xms2500m -cp \
/usr/local/share/pipelines/nlpp/modules/EHU-srl-server/IXA-EHU-srl-3.0.jar ixa.srl.SRLServer en
```

The command can be read from e.g. `/proc/29891/cmdline`. The script looks whether the string `SRLServer` appear in that line.

```
"bin/kill_eSRL_server" 2≡
      #!/bin/bash
      ⟨ define awk-script to extract eSRL process-id 1 ⟩
      procnum=`netstat -tulpn 2>/dev/null | grep 5005 | gawk $awkscript`
      grep SRLServer /proc/$procnum/cmdline
      res=$?
      if
          [ $res == 0 ]
      then
          echo process found: $procnum
          sudo kill $procnum
      else
          echo eSRL process not found
      fi
      ◇
```