PASTEBIN   new paste   trends   API   tools   faq   search…   Gu

VirtualBox API - Run Process in guest and read stdout

NOTEIRAK   FEB 24TH, 2015 (EDITED)   593   NEVER

SHARE

TWEET

Java   5.40 KB   raw   download   clone   embed   report   print

```java
/*
 *            DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
 *                    Version 2, December 2004
 *
 * Everyone is permitted to copy and distribute verbatim or modified
 *  copies of this license document, and changing it is allowed as long
 * as the name is changed.
 *
 *            DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
 *   TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
 *
 *  0. You just DO WHAT THE FUCK YOU WANT TO.
 */

import java.util.Arrays;
import org.virtualbox_4_3.GuestSessionStatus;
import org.virtualbox_4_3.IEvent;
import org.virtualbox_4_3.IEventListener;
import org.virtualbox_4_3.IEventSource;
import org.virtualbox_4_3.IGuestProcess;
import org.virtualbox_4_3.IGuestSession;
import org.virtualbox_4_3.IMachine;
import org.virtualbox_4_3.ISession;
import org.virtualbox_4_3.LockType;
import org.virtualbox_4_3.ProcessCreateFlag;
import org.virtualbox_4_3.ProcessWaitForFlag;
import org.virtualbox_4_3.ProcessWaitResult;
import org.virtualbox_4_3.VBoxEventType;
import org.virtualbox_4_3.VirtualBoxManager;

public class GuestSessionProcess {

    private static String errorIfNull(String systemProperty) {
        String value = System.getProperty(systemProperty);
        if (value != null) {
            return value;
        } else {
            throw new RuntimeException(systemProperty + " is not set");
        }
    }

    public static void main(String[] args) {
        String vmId = errorIfNull("vbox.vm");
        String guestUser = errorIfNull("vbox.guest.user");
        String guestPass = errorIfNull("vbox.guest.pass");
        String exec = errorIfNull("vbox.guest.exec");

        System.out.println("Starting");
        VirtualBoxManager vboxManager = VirtualBoxManager.createInstance(null);
        try {
            if (System.getProperty("vbox.ws") != null) {
                String host = System.getProperty("vbox.ws.host", "http://localhost:18083");
```

PASTEBIN   new paste   trends   API   tools   faq   search...   Gu

```
55.              vboxManager.connect(host, user, pass);
56.          }
57.          System.out.println("Connected");
58.          try {
59.              ISession session = vboxManager.getSessionObject();
60.              IMachine vm = vboxManager.getVBox().findMachine(vmId);
61.              vm.lockMachine(session, LockType.Shared);
62.              try {
63.                  System.out.println("Machine locked");
64.                  IGuestSession guestSess = session.getConsole().getGuest().createSession(guestUser, guestPass, null, null);
65.                  try {
66.                      System.out.println("Session created");
67.
68.                      guestSess.waitFor(1L, 30 * 1000L);
69.                      if (!guestSess.getStatus().equals(GuestSessionStatus.Started)) {
70.                          throw new RuntimeException("Guest session did not start after 30 sec");
71.                      }
72.                      IGuestProcess process = guestSess.processCreate(exec, null, null,
      Arrays.asList(ProcessCreateFlag.WaitForStdOut), 0L);
73.
74.                      IEventSource es = process.getEventSource();
75.                      IEventListener el = es.createListener();
76.                      es.registerListener(el, Arrays.asList(VBoxEventType.Any), false);
77.
78.                      try {
79.                          System.out.println("Guest process created");
80.                          ProcessWaitResult pwr = process.waitFor((long) ProcessWaitForFlag.Start.value(), 30 * 1000L);
81.                          System.out.println("Process wait result: " + pwr);
82.                          boolean keepLooping = true;
83.                          do {
84.                              IEvent ev = es.getEvent(el, 200);
85.                              if (ev != null) {
86.                                  es.eventProcessed(el, ev);
87.                              }
88.                              /* This is how you should normally do it, but waiting for Stdout is not currently implemented – see
      http://hyperbox.altherian.org/kb/guessProcessHandling.txt
89.                              ProcessWaitResult wr = process.waitForArray(Arrays.asList(ProcessWaitForFlag.StdOut,
      ProcessWaitForFlag.Terminate), 200L);
90.                              System.out.println("Process wait result: " + wr);
91.                               */
92.                              byte[] stdOut = process.read(1L, 64L, 0L);
93.                              System.out.print(new String(stdOut));
94.                              keepLooping = !process.getStatus().toString().contains("Terminated");
95.                          } while (keepLooping);
96.                          System.out.println("Process exit code: " + process.getExitCode());
97.                      } finally {
98.                          es.unregisterListener(el);
99.                          if (!process.getStatus().toString().contains("Terminated")) {
100.                             process.terminate();
101.                         }
102.                     }
103.                 } finally {
104.                     System.out.println("Session close");
105.                     guestSess.close();
106.                 }
107.             } catch (Throwable t) {
108.                 t.printStackTrace();
109.             } finally {
110.                 System.out.println("Machine unlock");
111.                 session.unlockMachine();
112.             }
113.         } finally {
114.             if (System.getProperty("vbox.ws") != null) {
```

PASTEBIN

new paste   trends   API   tools   faq   search...

```java
117.             System.out.println("Disconnected");
118.         }
119.     } finally {
120.         vboxManager.cleanup();
121.         System.out.println("Closing");
122.     }
123.   }
124.
125. }
```

RAW Paste Data

```java
            byte[] stdOut = process.read(1L, 64L, 0L);
            System.out.print(new String(stdOut));
            keepLooping = !process.getStatus().toString().contains("Terminated");
        } while (keepLooping);
        System.out.println("Process exit code: " + process.getExitCode());
    } finally {
        es.unregisterListener(el);
        if (!process.getStatus().toString().contains("Terminated")) {
            process.terminate();
```