

---

Subject: [BUG&FIX] LocalProcess can not be reused  
Posted by [dolik.rce](#) on Tue, 07 Sep 2010 19:29:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Jean-Paul found some unexpected behavior in the LocalProcess. It is easily reproducible:  
#include <Core/Core.h>  
using namespace Upp;

```
CONSOLE_APP_MAIN{  
    String app="mousepad";  
    LocalProcess p;  
    p.Start(app);  
    p.Kill();  
    p.Start(app);  
    p.Kill(); // <-- doesn't work, the app is continues running  
}
```

Fix is simple as well. The problem is, that `exit_code` is not reset after killing the process. So setting it back to null in `Free()` is enough to get correct behavior:

```
void LocalProcess::Free() {  
#ifdef PLATFORM_WIN32  
    if(hProcess) {  
        CloseHandle(hProcess);  
        hProcess = NULL;  
    }  
    if(hOutputRead) {  
        CloseHandle(hOutputRead);  
        hOutputRead = NULL;  
    }  
    if(hInputWrite) {  
        CloseHandle(hInputWrite);  
        hInputWrite = NULL;  
    }  
#endif  
#ifdef PLATFORM_POSIX  
    LLOG("\nLocalProcess::Free, pid = " << (int)getpid());  
    LLOG("rpipe[" << rpipe[0] << ", " << rpipe[1] << "]\n");  
    LLOG("wpipe[" << wpipe[0] << ", " << wpipe[1] << "]\n");  
    if(rpipe[0] >= 0) { close(rpipe[0]); rpipe[0] = -1; }  
    if(rpipe[1] >= 0) { close(rpipe[1]); rpipe[1] = -1; }  
    if(wpipe[0] >= 0) { close(wpipe[0]); wpipe[0] = -1; }  
    if(wpipe[1] >= 0) { close(wpipe[1]); wpipe[1] = -1; }  
    if(pid) waitpid(pid, 0, WNOHANG | WUNTRACED);  
    pid = 0;  
    output_read = false;  
#endif  
    exit_code = Null; // <-- Added  
}
```

}

Mirek, can you apply it, please?

Best regards,  
Honza

---