
Subject: LocalProcess child process killing in Windows

Posted by [koldo](#) on Tue, 07 Jul 2015 08:51:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek

Now LocalProcess works very well. However in windows, when a process is killed, child process are not so orphan processes remain working. It happens for example when launching .bat files.

Here it is TerminateChildProcesses(), that could be added into LocalProcess::Kill(), and searches and kills all child processes. As it uses a DWORD instead of a HANDLE, in DoStart() it would be necessary to get the process dwProcessId in addition to the hProcess.

```
#include <TIHelp32.h>
```

```
Vector<DWORD> GetChildProcessList(DWORD processId)
{
    // Gets all processes list
    PROCESSENTRY32 proc;
    Vector<DWORD> child, all, parents;
    HANDLE hSnap = CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS, 0);
    if (hSnap == INVALID_HANDLE_VALUE)
        return child;
    proc.dwSize = sizeof(proc);
    long f = Process32First(hSnap, &proc);
    while (f) {
        all << proc.th32ProcessID;
        parents << proc.th32ParentProcessID;
        f = Process32Next(hSnap, &proc);
    }
    CloseHandle(hSnap);

    // Gets process child list
    child << processId;
    int init = 0;
    while (true) {
        int count = child.GetCount();
        if (init >= count)
            break;
        for (int cid = init; cid < count; ++cid) {
            for (int i = 0; i < all.GetCount(); ++i) {
                if (parents[i] == child[cid])
                    child << all[i];
            }
        }
        init = count;
    }
    child.Remove(0);
}
```

```
    return child;
}

void TerminateChildProcesses(DWORD dwProcessId, UINT uExitCode)
{
    Vector<DWORD> child = GetChildProcessList(dwProcessId);
    for (int i = 0; i < child.GetCount(); ++i) {
        HANDLE hProcess = OpenProcess(PROCESS_TERMINATE, FALSE, child[i]);
        TerminateProcess(hProcess, uExitCode);
    }
}
```
