Forcing a file handle closed when it has been opened remotely

devblogs.microsoft.com/oldnewthing/20130916-00

September 16, 2013

Raymond Chen

Today's Little Program closes a file handle that was opened remotely. It builds on <u>previous discussion on how to use the `NetXxx` functions</u>.

```
int __cdecl wmain(int argc, wchar_t **argv)
{
 FILE_INFO_3 *pinfo3;
 NET_API_STATUS status;
 DWORD_PTR resumeHandle = 0;
 do {
  DWORD actual, estimatedTotal;
  status = NetFileEnum(NULL, NULL, NULL, 3,
                       (LPBYTE*)&pinfo3,
                       MAX_PREFERRED_LENGTH,
                       &actual,
                       &estimatedTotal,
                       &resumeHandle);
  if (status == NERR_Success ||
      status == ERROR_MORE_DATA) {
   for (DWORD i = 0; i < actual; i++) {
    if (lstrcmpiW(argv[1], pinfo3[i].fi3_pathname) == 0) {
     wprintf(L"Closing %ls result %d\n", pinfo3[i].fi3_pathname,
             NetFileClose(NULL, pinfo3[i].fi3_id));
     status = ERROR_NO_MORE_FILES;
     break;
    }
   }
   NetApiBufferFree(pinfo3);
  }
 } while (status == ERROR_MORE_DATA);
 return 0;
}
```

Forcing a network file handle closed does *not* actually close the handle. This makes it very different from <u>the various "force handle closed" utilities out there</u>. Rather, forcing a network file handle closed is accomplished by simulating a network failure, so that when the remote machine tries to use the handle again, it's told, "Wha? I'm sorry, we must have a bad

connection, because I'm not sure what you're talking about." Since programs which access network resources must deal with the possibility of network connectivity loss, this deception does not violate the interface contract.

(Doing this to handles to local resources is a much riskier undertaking, because applications expect access to local files to remain valid for the lifetime of the handle. There is no equivalent of transient network connectivity failure for local files on non-removable drives. There is also no API for simulating it.)

Raymond Chen

Follow