

Wt - Bug #6342

Leak in Wt::Http::Request::parseFormUrlEncoded

04/16/2018 04:02 PM - Emeric Poupon

Status:	Feedback	Start date:	04/16/2018
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			

Description

Hello,

Using Wt 4.0.2, wthttpd, it seems I have a leak that is not related to my code.

I have some custom resources that call Wt::Http::Request::getParameter though.

```
==27946== HEAP SUMMARY:
==27946==    in use at exit: 606,612 bytes in 1,223 blocks
==27946== total heap usage: 139,943 allocs, 138,720 frees, 461,500,696 bytes allocated
==27946==
==27946== 277,517 (88 direct, 277,429 indirect) bytes in 1 blocks are definitely lost in loss reco
rd 164 of 164
==27946==    at 0x4C2C21F: operator new(unsigned long) (vg_replace_malloc.c:334)
==27946==    by 0x8C68C1E: allocate (new_allocator.h:104)
==27946==    by 0x8C68C1E: allocate (alloc_traits.h:436)
==27946==    by 0x8C68C1E: _M_get_node (stl_tree.h:505)
==27946==    by 0x8C68C1E: _M_create_node<const std::piecewise_construct_t&, std::tuple<const std:
: __cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> >&>, std::tuple<> > (stl
_tree.h:559)
==27946==    by 0x8C68C1E: std::_Rb_tree_iterator<std::pair<std::__cxx11::basic_string<char, std::
char_traits<char>, std::allocator<char> > const, std::vector<std::__cxx11::basic_string<char, std:
:char_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::basic_string<char, std::c
har_traits<char>, std::allocator<char> > > > > std::_Rb_tree<std::__cxx11::basic_string<char, st
d::char_traits<char>, std::allocator<char> >, std::pair<std::__cxx11::basic_string<char, std::char
_traits<char>, std::allocator<char> > const, std::vector<std::__cxx11::basic_string<char, std::cha
r_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::basic_string<char, std::char_
traits<char>, std::allocator<char> > > > >, std::Select1st<std::pair<std::__cxx11::basic_string<c
har, std::char_traits<char>, std::allocator<char> > const, std::vector<std::__cxx11::basic_string<
char, std::char_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::basic_string<ch
ar, std::char_traits<char>, std::allocator<char> > > > >, std::less<std::__cxx11::basic_string<c
har, std::char_traits<char>, std::allocator<char> > >, std::allocator<std::pair<std::__cxx11::basi
c_string<char, std::char_traits<char>, std::allocator<char> > const, std::vector<std::__cxx11::bas
ic_string<char, std::char_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::bas
ic_string<char, std::char_traits<char>, std::allocator<char> > > > > > ::_M_emplace_hint_unique<std
::piecewise_construct_t const&, std::tuple<std::__cxx11::basic_string<char, std::char_traits<char>
, std::allocator<char> > const&>, std::tuple<> > (std::_Rb_tree_const_iterator<std::pair<std::__cx
x11::basic_string<char, std::char_traits<char>, std::allocator<char> > const, std::vector<std::__cx
x11::basic_string<char, std::char_traits<char>, std::allocator<char> >, std::allocator<std::__cx
x11::basic_string<char, std::char_traits<char>, std::allocator<char> > > > > >, std::piecewise_const
ruct_t const&, std::tuple<std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<
char> > const&&&, std::tuple<>&&) [clone .isra.156] (stl_tree.h:2196)
==27946==    by 0x8C6927B: operator[] (stl_map.h:483)
==27946==    by 0x8C6927B: Wt::Http::Request::parseFormUrlEncoded(std::__cxx11::basic_string<char,
 std::char_traits<char>, std::allocator<char> > const&, std::map<std::__cxx11::basic_string<char,
 std::char_traits<char>, std::allocator<char> >, std::vector<std::__cxx11::basic_string<char, std::
char_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::basic_string<char, std::ch
ar_traits<char>, std::allocator<char> > > >, std::less<std::__cxx11::basic_string<char, std::char_
traits<char>, std::allocator<char> > >, std::allocator<std::pair<std::__cxx11::basic_string<char,
 std::char_traits<char>, std::allocator<char> > const, std::vector<std::__cxx11::basic_string<char,
 std::char_traits<char>, std::allocator<char> >, std::allocator<std::__cxx11::basic_string<char, s
td::char_traits<char>, std::allocator<char> > > > > > > &) (Request.C:401)
==27946==    by 0x8CC6D8B: Wt::CgiParser::parse(Wt::WebRequest&, Wt::CgiParser::ReadOption) (CgiPa
```

```

rser.C:235)
==27946==   by 0x8CF4766: Wt::WebController::handleRequest(Wt::WebRequest*) (WebController.C:595)
==27946==   by 0x7F4A7DE: __invoke_impl<void, void (Wt::WebController::* const&)(Wt::WebRequest*)
, Wt::WebController*&, http::server::HTTPRequest*&> (functional:227)
==27946==   by 0x7F4A7DE: __invoke<void (Wt::WebController::* const&)(Wt::WebRequest*), Wt::WebCo
ntroller*&, http::server::HTTPRequest*&> (functional:251)
==27946==   by 0x7F4A7DE: operator()<Wt::WebController*&, http::server::HTTPRequest*&> (functiona
l:604)
==27946==   by 0x7F4A7DE: __call<void, 0ul, 1ul> (functional:934)
==27946==   by 0x7F4A7DE: operator()<> (functional:993)
==27946==   by 0x7F4A7DE: asio_handler_invoke<std::Bind<std::Mem_fn<void (Wt::WebController::*)
(Wt::WebRequest*)>(Wt::WebController*, http::server::HTTPRequest*)>> (handler_invoke_hook.hpp:69)
==27946==   by 0x7F4A7DE: invoke<std::Bind<std::Mem_fn<void (Wt::WebController::*) (Wt::WebReque
st*)>(Wt::WebController*, http::server::HTTPRequest*)>, std::Bind<std::Mem_fn<void (Wt::WebContr
oller::*) (Wt::WebRequest*)>(Wt::WebController*, http::server::HTTPRequest*)>> (handler_invoke_hel
pers.hpp:37)
==27946==   by 0x7F4A7DE: boost::asio::detail::completion_handler<std::Bind<std::Mem_fn<void (W
t::WebController::*) (Wt::WebRequest*)> (Wt::WebController*, http::server::HTTPRequest*)>>::do_com
plete(boost::asio::detail::task_io_service*, boost::asio::detail::task_io_service_operation*, boost
::system::error_code const&, unsigned long) (completion_handler.hpp:68)
==27946==   by 0x8A0F4B7: complete (task_io_service_operation.hpp:38)
==27946==   by 0x8A0F4B7: do_run_one (task_io_service.ipp:372)
==27946==   by 0x8A0F4B7: run (task_io_service.ipp:149)
==27946==   by 0x8A0F4B7: run (io_service.ipp:59)
==27946==   by 0x8A0F4B7: Wt::WIOService::run() (WIOService.C:180)
==27946==   by 0x91A0E6E: ??? (in /usr/lib/x86_64-linux-gnu/libstdc++.so.6.0.22)
==27946==   by 0x9BB8493: start_thread (pthread_create.c:333)
==27946==   by 0x9EB6AFE: clone (clone.S:97)
==27946==
==27946== LEAK SUMMARY:
==27946==   definitely lost: 88 bytes in 1 blocks
==27946==   indirectly lost: 277,429 bytes in 34 blocks
==27946==   possibly lost: 0 bytes in 0 blocks
==27946==   still reachable: 329,095 bytes in 1,188 blocks
==27946==   suppressed: 0 bytes in 0 blocks
==27946== Reachable blocks (those to which a pointer was found) are not shown.
==27946== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==27946==
==27946== For counts of detected and suppressed errors, rerun with: -v
==27946== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)

```

History

#1 - 06/22/2018 10:15 AM - Roel Standaert

- Status changed from New to Feedback

Do you have some example code or something? I checked a few examples, like the te-benchmark example, and those don't seem to report any leaks.