Практическая работа 6

DEFER_AFTER_NULL.pdf

Задание

Location: /server/sql/opt_split.cc:1249	Undecided
Function	_ZN4JOIN26inject_best_splitting_condEy
Warning message	After having been compared to a NULL value at opt_split.cc:1246, pointer 'inj_cond' is dereferenced at opt_split.cc:1249.

```
1244      return true;
1245  }
1246  if (inj_cond)
1247    inj_cond->fix_fields(thd,0);
1248
```

3 of 5

```
1249  if
      (inject_cond_into_where(inj_cond->copy_andor_structure(thd)
      ))
1250    return true;
1251
1252  select_lex->uncacheable|= UNCACHEABLE_DEPENDENT_INJECTED;
1253  st_select_lex_unit *unit= select_lex->master_unit();
1254  unit->uncacheable|= UNCACHEABLE_DEPENDENT_INJECTED;
```

Анализ кода

Этот Warning анализатора относится к **CWE-476: NULL PointerDereference**.

Рассмотри данные фрагмент кода. Указатель *inj_cond* на 1246 строчке сравнивается с NULL. Если он не равен NULL, то у *inj_cond* вызывается метод *fix_fields(thd, 0)*.

Далее в следующем условии (не вложенном) на 1249 строчке вызывается функция *inject_cond_into_where* и в нее передается результат вызова метода *copy_andor_structure(thd)* у *inj_cond*. В этот момент происходит разыменование указателя.

Скорее всего, анализатор подумал, что это разыменование происходит в else блоке предыдущего условия. В таком случае *inj_cond* был бы равен NULL.

Но на самом деле это не так. Однако ошибка все равно присутствует. Потенциально, *inj_cond* может быть равен NULL. В момент разыменования на 1249 строчке.

Предложение по исправлению

На 1249 строчке нужно добавить проверку на NULL перед разыменованием. То есть:

```
if (inj_cond)
    inj_cond->fix_fields(thd, 0);

if (inj_cond && inject_cond_into_where(inj_cond->copy_andor_structure(thd))) {
    return true;
}
```

Location: /server/sql/opt_split.cc:1249	Should fix
Function	_ZN4JOIN26inject_best_splitting_condEy
Warning message	After having been compared to a NULL value at opt_split.cc:1246, pointer 'inj_cond' is dereferenced at opt_split.cc:1249.
Комментарий	на 1249 строчке вызывается функция *inject_cond_into_where* и в нее передается результат вызова метода *copy_andor_structure(thd)* у *inj_cond*. Потенциально разыменовывается пустой указатель

```
1244      return true;
1245  }
1246  if (inj_cond)
1247    inj_cond->fix_fields(thd,0);
1248
```

```
1249  if
      (inject_cond_into_where(inj_cond->copy_andor_structure(thd)
      ))
```

```
1250    return true;
1251
1252  select_lex->uncacheable|= UNCACHEABLE_DEPENDENT_INJECTED;
1253  st_select_lex_unit *unit= select_lex->master_unit();
1254  unit->uncacheable|= UNCACHEABLE_DEPENDENT_INJECTED;
```