MINISTERSTVO
INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA
A INFORMATIZÁCIE
SLOVENSKEJ REPUBLIKY

CSIRT.SK

Zraniteľnosť Nette Framework

Vypracoval: CSIRT.SK
Ministerstvo investícií, regionálneho rozvoja a informatizácie SR
Pribinova 25
811 09 Bratislava

Dátum vypracovania správy: Október 2024

TLP: Clear

Bezpečnostní analytici CSIRT.SK objavili potenciálnu zraniteľnosť vo webovom frameworku Nette, ktorá umožňuje vykonávať útoky typu SQL injection.

Zraniteľnosť sa nachádza v knižnici Database a súvisí s absenciou ošetrovania používateľských vstupov, ktoré preberá funkcia *where($by)*. Táto metóda podporuje filtrovanie výsledkov databázovej požiadavky podľa poľa, v ktorom kľúče predstavujú názvy stĺpcov a hodnoty daných kľúčov predstavujú požadované hodnoty. Problematickými sú kľúče párov premenných *{"kľúč":"hodnota"}*.

Útočník môže vykonať útok typu SQL injection podvrhnutím kľúča premennej priamo v URL v prehliadači alebo v závislosti od konkrétneho prípadu aj vo vstupoch aplikácie.

V prípade, že sa do metódy where odošle PHP array v tvare *{"SQLi payload"=>"hodnota"}*, kde ***SQLi payload*** je hodnota zadaná používateľom, daný **payload** sa vykoná v databáze, čím je možné docieliť získavanie citlivých údajov. Príklad útoku:

```
http://localhost:8000/?id%3d1)+UNION+SELECT+2,version()+FROM+test+WHERE+(1=1
```

Tým sa v Nette Framework vyskladá nasledovná databázová požiadavka:

```
"SELECT id, name FROM test (WHERE ". $column . " = ?)"
```

Po doplnení hodnoty za premennú *column* dostaneme:

```
"SELECT id, name FROM test (WHERE ". "id=1) UNION SELECT 2,version() FROM test WHERE (1" . " = ?)"
```

A teda finálna parametrizovaná požiadavka na databázu bude vyzerať:

```
"SELECT id, name FROM test (WHERE id=1) UNION SELECT 2,version() FROM test WHERE (1= ?)"
```

Zraniteľnosť sa vo webovej aplikácii nachádza, pokiaľ využíva knižnicu Nette/Database a ak je v Presenteri použitý kód:

```
$httpRequest = $this->getHttpRequest();

$filter = $httpRequest->getQuery();

$this->template->terms = $this->testFacade->findBy($filter);
```

Zároveň Facade používa metódu *findBy* definovanú nasledovne (preberá hodnotu z funkcie *where*):

```
public function findBy(array $by)  {

   return $this->database->table($this->tableName)->where($by);

 }
```

Možné škody:

- **Únik citlivých informácií**

Odporúčania:

Pokiaľ pri vývoji webovej aplikácie v Nette Framework implementujete danú zraniteľnú funkciu, odporúčame dodatočne ošetriť používateľské vstupy (resp. kľúč premennej) v rámci best practice bezpečného vývoja.