

Testovanie a dynamická analýza

Projekt: Testovanie funkcie `pear_mail`

1 Cieľ

Otestovať funkciu `pear_mail` v súbore `ex_mail.php`. Funkcia prijíma 6 argumentov a na základe ktorých posiela email. Primárne sa používa na odoslanie informácii o objednávke.

2 Technológie

Bol použitý testovací framework **PHP Unit** ktorý poskytuje rozhranie na vytváranie testovacích sád na automatické testovanie funkcií a tried. Používa asserty na overenie, že správanie jednotky kódov v rámci testu sa chová ako sa očakávalo.

3 Metoda

Ako kritérium pokrytia bolo vybrané BCC(Base Choice Coverage) - Kritérium pokrytia bazových blokov, ktorá funguje na modely vstupných domén.

4 Testovacia sada

Bolo vytvorených 5 nasledovných charakteristik:

1. Vstup HTML
 - **1**
 - Všetko ostatné
2. Vstup from
 - prázdny
 - **neprázdny**
3. proforma - Návratova hodnota z DB
 - **1**
 - **0**
 - `array()` - prázdne pole
4. Simulovanie chyby metódy `send` v triede `Mail`
 - **TRUE**
 - **FALSE**

5. Nadpis

- **Začína s Objednávka č. a za ním je číslo**
- Začína s Objednávka č a následne znakom ktorý není bodka
- Za 2. medzerou je číslo
- Za 2. medzerou není číslo
- Obsahuje menej ako jednu medzeru

Zvýraznene položky sú bazovými blokmi. Na základe charakteristik bolo vytvorených 10 testovacích sád. Prvá testovacia sada je kombinácia všetkých bazových blokov a ostatné sú kombinácia každého nebazového bloku s ostatnými bazovými blokmi.

5 Dosiahnuté výsledky

Bola pokrytá metóda BCC, taktiež 100% pokrytie riadkov kódu. Celkovo bolo nájdených 4 chyby. 2 z toho spôsobujú pád programu, jedna z externej závislosti ktorá môže byť ošetrená v danej metóde. Ďalšie 2 generujú zle telo správy. Ďalšie chyby sa môžu nachádzať v zlej konfigurácii servera, globalnej premennej \$CONFIG.

6 Najdené chyby

Verzia:

- OS - Ubuntu 14.04 64bit
- PHP - 5.5.9-1
- ex_mail - 1.1

6.1 Pád programu pri neošetrenom nadpise

- Popis: Program neošetruje nadpis z parametru, pri nevhodnom nadpise sa program pokúsi pristupovať k offsetu pola ktorý neexistuje.
- Kroky pre reprodukciu:
 1. Ako 3. parameter(\$nadpis) predať string ktorý bude obsahovať menej ako 2 medzery, napríklad Nadpis 10
 2. Zavolať funkciu
- Očakávané chovanie: Funkcia nepošle email a vráti false
- Riešenie: Po zavolaní funkcie explode kontrolovať veľkosť pola \$zjisti_cislo_obj

```
if (count($zjisti_cislo_obj) < 3) return false;
```

6.2 Pád programu pri nenájdení výsledku z databáze

- Popis: Program neošetruje výstup z databáze a môže pristupovať do neinicializovaného pola
- Kroky pre reprodukciu:

1. Do nadpisu vložit identifikačné číslo objednávky ktoré neexistuje, napr. Objednávka č. -1
 2. Zavolať funkciu
- Očakávané chovanie: ID objednávky neexistuje, email sa neodošle a funkcia vráti false
 - Riešenie: Kontrolovať výstup z databáze.

```
if (empty($prof_ano_ne)) return false;
```

6.3 Pod'akovacia správa

- Popis: Pod'akovacia správa sa môže pripájať aj keď sa nejedná o objednávku, pretože `strcmp` porovnáva jednotlivé bajty a v nadpise sa nachádzajú aj lokalizované znaky ktoré sú multi-bajtové.
- Kroky pre reprodukciu:
 1. Do nadpisu vložit Objednávka č, 10
 2. Zavolať funkciu
- Očakávané chovanie: Pod'akovacia správa by sa nemala pripojiť k telu správy
- Riešenie: Keďže PHP nemá vstavanú funkciu na porovnávanie multi-byte stringov, tak buď naprogramovať vlastnú alebo použiť následovný workaround a nahradiť konštantu 13

```
if (strcmp($nadpis, "Objednávka č.", mb_strlen('Objednávka č.')) == 0)
```

6.4 Proforma

- Popis: Údaje o platbe sa nevkladajú do tela správy, pretože v hlavičke html správy je `id='ObsahMailu'>` a v `str_replace` sa očakáva `id='obsahMailu'>`
- Kroky pre reprodukciu:
 1. Do nadpisu vložit také ID z DB aby `proforma == 1`
 2. Zavolať funkciu
- Očakávané chovanie: Do tela správy sa vloží text proformy.
- Riešenie: 2 možné:
 1. Upraviť hlavičku html na malé 'o'
 2. Upraviť `str_replace` na veľké 'O'

```
$body = str_replace('id="ObsahMailu">', 'id="ObsahMailu">' . ...
```