Προστασία και Ασφάλεια Υπολογιστικών Συστημάτων EAPINO 2018 **Project #1**

Λάμπρου Ιωάννης
1115201400088

Στεφανίδης - Βοζίκης Κωνσταντίνος
1115201400192

26 Απριλίου 2018

Defence

CSRF

Όσον αφορά την άμυνα για τις επιθέσεις CSRF. Ποστατευθήκαμε βάζοντας CSRF tokens σε κάθε φόρμα ή οποία προκαλεί αλλαγές στην σελίδα. Σε κώδικα PHP ή άμυνα μοιάζει κάπως έτσι:

```php
if (empty($_SESSION['token'])) {
    if (function_exists('mcrypt_create_iv')) {
        $_SESSION['token'] = bin2hex(mcrypt_create_iv(32, MCRYPT_DEV_URANDOM));
    } else {
        $_SESSION['token'] = bin2hex(shell_exec("openssl rand -base64 32"));
    }
}
$token = $_SESSION['token'];

if (visible_module($module_id)) {
  $message = $langDeactivate;
  $mod_activation = "
  <form id='myform".htmlspecialchars($module_id)."' style='display:inline;' action='".htmlspeci
    <a href='javascript:;' onclick=\"document.getElementById('myform".htmlspecialchars($module_
    ".htmlspecialchars($langDeactivate)."</a>
    <input type='hidden' name='eclass_module_id' value='".htmlspecialchars($module_id)."' />
    <input type='hidden' name='hide' value='0'/>
    <input type='hidden' name='token' value='$token'/>
  </form>
  ";
```

Πρώτα ενεργοποιούμε ενα token και κατόπιν το βάζουμε ως hidden field στην φόρμα που μας ενδιαφέρει. Ο έλεγχος εγκυρότητας γίνεται ως εξης:
if (isset($_POST['hide']) and $_POST['hide'] == 0 and !empty($_POST['token']) and (strcmp($_SESSION['token'], $_POST['token']) === 0))
Η ίδια λογική άμυνας υπάρχει σε κάθε φόρμα της ιστοσελίδας (η οποία προκαλεί αλλαγές). Επίσης, επειδή δεν υπάρχει άμυνα έναντι επιθέσεων CSRF σε GET

requests, διάφορα GET που άλλαζαν την σελίδα αλλάχθηκαν σε POST ώστε να γίνει η ίδια άμυνα. Ένα παράδειγμα είναι η διαγραφή χρήστη από την σελίδα του admin.

Attack

(α′) Σωστό. Το $\omega()$ είναι αυστηρότερο, άρα και η $f(x)$ θα είναι και $\Omega(g(n))$ (Αφού το Ω είναι υπερσύνολο του ω)

(β′) Λάθος. Για $n > 1$, πάντα η$(10n^2 + kn + c)$ θα είναι μεγαλύτερη ή ίση από τη $(4n^2 + 5n - 9)$ $(4n^2 + 5n - 9) = O(10n^2)$

(γ′) Σωστό. Ξέρουμε από τις διαφάνειες πως $\log n! = O(n \log n)$ (Άσκηση 4, σελ 15), ενώ $\log n! = \log 1 + \log 2 + ... + \log n >=$

$\log \frac{n}{2} + \log \frac{n}{2} + 1 + ... + \log n = \log \frac{n}{2} * \frac{n}{2} = \frac{n \log n}{2} - \frac{n \log 2}{2} =>$

$\log n! >= \frac{n \log n}{2} - \frac{n \log 2}{2}$, Άρα και $\log n!$ =$\Omega(n \log n)$. Αφού το $\log n!$ είναι και O και Ω του $(n \log n)$, τότε θα είναι και Θ

(δ′) Λάθος. Ξέρουμε πως $f(n) + g(n)$ =$\Omega(min(g(n), f(n)))$ ενώ $f(n) + g(n) = O(max(g(n), f(n)))$ άρα ισχύει μόνο στην περίπτωση που $f(n) = g(n)$

(ε′) Σωστό. Αν πάρουμε το όριο, $lim_{n\to\infty} \frac{n+2\sqrt{n}}{n\sqrt{n}} = lim_{n\to\infty} \frac{\sqrt{n}+2}{n} = 0$, άρα είναι O

(ϛ′) Σωστό. Ξέρουμε ότι το όριο $lim_{n\to\infty}(g(n) - f(n)) = -\infty$. Αν πάρουμε το όριο, $lim_{n\to\infty} \frac{2^{g(n)}}{2^{f(n)}} = lim_{n\to\infty} 2^{g(n)-f(n)}$ το οποίο και θα κάνει μηδέν, λόγω του αρχικού ορίου. Άρα και το $2^{f(n)}$ θα είναι $\Omega(2^{g(n)})$ άρα και ω.

(ζ′) Σωστό. Αποδείχτηκε στο προηγούμενο ερώτημα.

(η′) Σωστό. Έστω $f(x)$ =ω$(g(x)$, τότε θα πρέπει για $x >= x_0 f(x) > g(x)$. Ομοίως, αν $f(x)$ =$o(g(x)$, τότε θα πρέπει για $x >= x_0 f(x) < g(x)$ Έτσι, βλέπουμε ότι τα δύο αυτά ενδεχόμενα είναι ξένα.