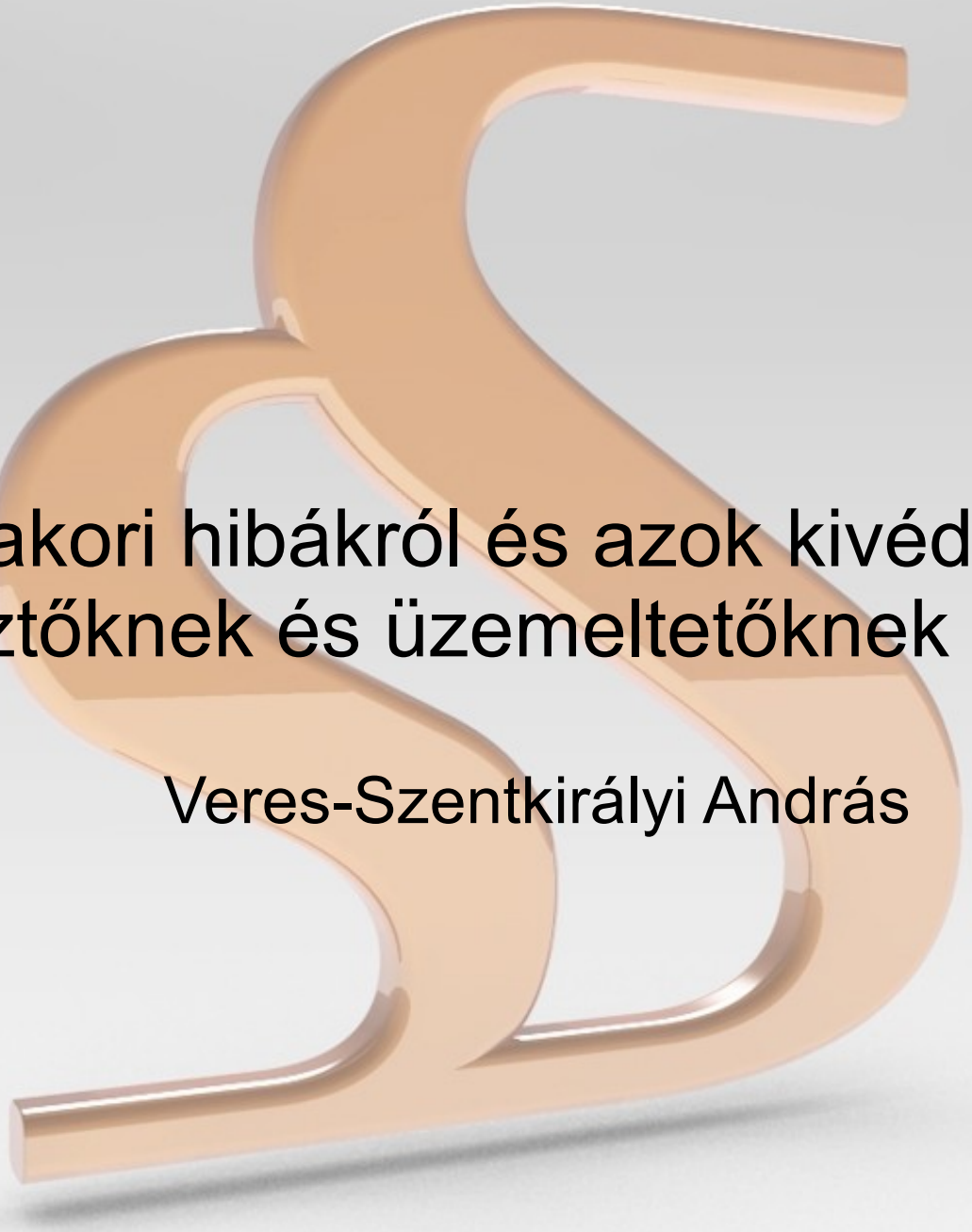
SILENT SIGNAL

Webapp (in)security



Gyakori hibákról és azok kivédéséről fejlesztőknek és üzemeltetőknek egyaránt

Veres-Szentkirályi András

SILENT SIGNAL

Rövid áttekintés

- Webalkalmazások fejlesztése során elkövetett leggyakoribb hibák ismertetése és elemzése
  - SQL injection
  - XSS
  - CSRF
- Hibák kihasználásának bemutatása
- Néhány példa arra, hogyan kerüljük el a hasonló helyzeteket

SILENT SIGNAL

XSS

- Aka. Cross-Site Scripting
- Kliensoldalról származó adat kerül a szerver által generált kimenetbe
- Három változat
  - DOM-alapú (helyi)
  - **Reflective**
  - **Persistent**

SILENT SIGNAL

Reflective XSS

- Klienstől érkező paraméterek felhasználása megfelelő szűrés nélkül...
  - egyszer
  - és csak a kliens munkamenetében
- Nem (tűnik) annyira veszélyes(nek)
- Phishingre kiváló

Reflective XSS

- **https://bank.com/search.php?search=<script>window.location='http://evil.com'</script>**
- **Egy kicsit kevésbé feltűnő módon:**

  https://bank.com/search.php?search=%3c%73%63%72%69%70%74%3e%77%69%6e%64%6f%77%2e%6c%6f%63%61%74%69%6f%6e%3d%27%68%74%74%70%3a%2f%2f%65%76%69%6c%2e%63%6f%6d%27%3c%2f%73%63%72%69%70%74%3e%3c%73%63%72%69%70%74%3e%77%69%6e%64%6f%77%2e%6c%6f%63%61%74%69%6f%6e%3d%27%68%74%74%70%3a%2f%2f%65%76%69%6c%2e%63%6f%6d%27%3c%2f%73%63%72%69%70%74%3e

SILENT SIGNAL

Tárolt XSS

- Klienstől érkező paraméterek felhasználása megfelelő szűrés nélkül...
  - majd tárolása szerveroldalon
  - ahonnan elérhető mindenki számára

XSS támadások

- Phishing
- Drive-by exploitok
- Deface
- Helyi hálózat elleni támadások
  - Lásd Linksys WPA kikapcsolás
- Ugródeszka CSRF-hez
  - Később még beszélünk róla
- XSS férgek

XSS férges esetek

- 2007. január 31.: Samy – MySpace
  - 20 óra alatt több mint egymillió felhasználó
- 2008. június 28.: Justin.tv
- 2009. szeptember 28.: Reddit.com
- Következő áldozat: Yahoo Meme?
  - http://www.hackersblog.org/2009/10/11/i-can-predict-the-future/

SILENT SIGNAL

XSS elleni védelem

- Bemenet szűrése
  - Tilos az <á>!
    - Biztosan elegendő?
    - Nem feltétlenül...
  - Használhatóság és biztonság közti egyensúly
    - Lassan kezdünk hozzászokni
- Kimenet szűrése
  - Sajnos még nem gyakori...
  - htmlspecialchars() a barátod

SILENT SIGNAL

SQL injection

- Példa: egyszerű azonosítás
- Felhasználónév + jelszó
- Megnézzük, hogy van-e olyan rekord, amely a megfelelő felhasználónevet és jelszót tartalmazza
  - SELECT * FROM users WHERE name=? AND password=?

Ami a háttérben történik

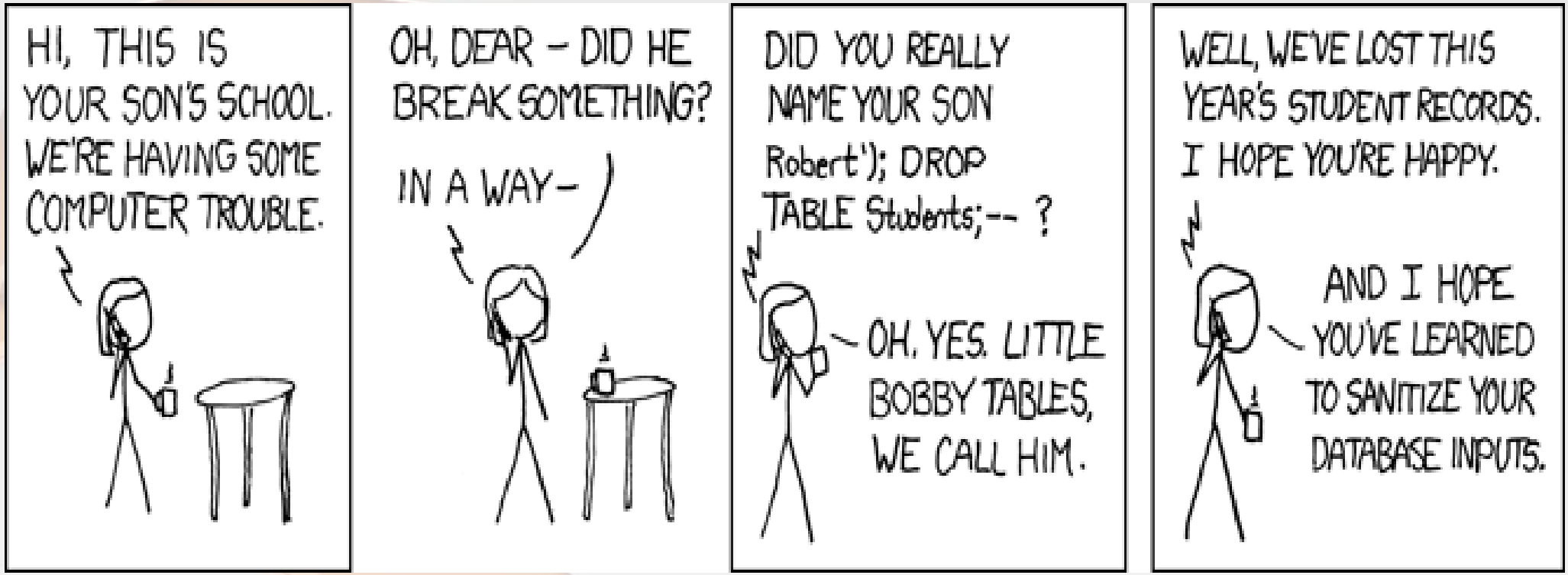
- Paraméter kiolvasása
- Lekérdezés összeállítása
  - SELECT * FROM users WHERE name='***admin***' AND password='***szupertitkosjelszó***'
- Mi történik, ha...
  - SELECT * FROM users WHERE name='***admin' #*** ' AND password=''

Mire van még lehetőségünk?

- Más adatok kiolvasása
  - UNION SELECT
- INSERT, UPDATE, DELETE utasítások
  - Adatmódosító lekérdezések injectelésével
  - Query Stacking
- Fájlrendszer olvasása (!)
- Parancsvégrehajtás (!!)

SILENT SIGNAL

Query stacking



SILENT SIGNAL

SQLi elleni védelem

- **Szűrjük a bemeneteket!**
  - A számok legyenek valóban számok!
  - A stringhatárolókat escape-eljük!
- **...illetve méginkább: használjunk előkészített lekérdezéseket és/vagy ORM keretrendszert!**
- **Best Practices**
  - Hasheljük a jelszavakat!
  - A DBMS a szükséges legalacsonyabb privilégiumokkal fusson!
  - Legyen külön saját adatbázisa minden alkalmazásnak!

A böngésző szépségei

- Többlapos böngészés FTW!
  - Sok munkamenet ...
  - … különböző oldalakon ...
  - … ugyanabban a böngészőben
- Sütik
  - Kliensoldali adattárolás
- Munkamenet-azonosítók
  - (remélhetőleg) véletlen(szerű) azonosítók
  - Legtöbbször kliensoldalon tárolva

SILENT SIGNAL

Same-origin Policy

- „... permits scripts running on pages originating from the same site to access each other's methods and properties with no specific restrictions — but prevents access to most methods and properties across pages on different sites.”
- A Web 2.0 világban néha valóban szükség lehet, hogy más oldal is elérje az adatainkat
  - Erre itt az AJAX
  - Ami még nem sérti az SoP-t!

Cross-Site Request Forgery

- Sima HTTP kéréseket küldhetünk más oldalaknak egy weboldalról
  - IFRAME-ek, képek, stb...
  - JavaScript
- A kéréssel együtt a böngésző elküldi az adott szervernek szóló sütiket
  - A kérés a meglévő munkamenetünkhöz tartozik
- Innentől szabad a pálya...

SILENT SIGNAL

CSRF védelem

- Űrlapadatok védelme tokennel
  - A támadó nem ismeri az értékét

  1) Token generálás
  2) Token tátolás (DB)
  3) Rejtett mező minden HTML űrlapon
  4) Csak megfelelő tokennel rendelkező adatok elfogadása
- Referer ellenőrzés

További információk

- Open Web Application Security Project: http://www.owasp.org
- XSSed: http://www.xssed.com
- SQL injection cheat sheet: http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/

SILENT SIGNAL

Köszönöm a figyelmet!

Kérdések?

vsza@silentsignal.hu
http://www.silentsignal.hu