# 1DI1526 - Ochrona danych w systemach informatycznych

## Lab 7 Projekt -- prosta bezpieczna aplikacja internetowa

*Ostatnia modyfikacja: B. Sawicki 23.11.2021 08:33*

### Cel zajęć

Celem zajęć jest napisanie prostej aplikacji internetowej spełniającej wysokie standardy bezpieczeństwa.

### Wprowadzenie

Tworząc oprogramowanie programiści najczęściej skupiają się na funkcjonalności i wydajności systemu. Jednak tym razem pełna uwaga powinna być skupiona na kwestiach bezpieczeństwa. Dlatego realizując zadanie nie ma większego znaczenia wygląda aplikacji, ani jej możliwości - liczy się to w jaki sposób sprawdza ona poprawność danych, czy ma restrykcyjne ustawienia początkowe, czy poprawnie wykorzystuje algorytmy kryptograficzne.

Dobrym materiałem do tego rodzaju ćwiczeń jest moduł uwierzytelniania. Jest on kluczowym elementem ochrony systemu, dlatego powinien być napisany wyjątkowo starannie. Na poprzednich zajęciach uczyliśmy się wielu technik, które powinny być wykorzystane przy tej okazji:

- walidacja danych wejściowych (z negatywnym nastawieniem),
- opóźnienia i limit prób (żeby utrudnić zdalne zgadywanie i atak brute-force),
- ograniczone informowanie o błędach (np. o tym przyczynie odmowy uwierzytelenia),
- bezpieczne przechowywanie hasła (wykorzystanie kryptograficznych funcji mieszających, wykorzystanie soli, wielokrotne hashowanie)
- kontrola siły hasła, żeby uświadomić użytkownikowi problem
- monitorowanie pracy systemu (np. żeby poinformować użytkownika o nowych komputerach, które łączyły się z jego kontem)
- zarządzanie uprawnieniami do zasobów

Za modułem uwierzytelniania napiszmy prostą funkcjonalność przechowywania i współdzielenia haseł do innych serwisów. Aplikacja umożliwia kontrolę dostępu do haseł: hasła prywatne, albo dostępne dla wybranych użytkowników.

### Potrzebna wiedza

- podstawowa znajomość języka python
- wiedza i umiejętności z poprzednich zajęć dotyczących konfiguracji Apache oraz bezpieczeństwa aplikacji internetowych,

### Dodatkowe informacje

- http://www.petefreitag.com/item/505.cfm - 20 ways to Secure your Apache

Configuration

- http://httpd.apache.org/docs/2.2/misc/security_tips.html - Apache Security Tips

Hasła dla Google: authentication, WWW security

**Podstawowe wymagania:**

Napisz aplikację WWW realizującą uwierzytelnianie w oparciu o tajne hasło. Zwróć uwagę na:

- **(niezbędne)** restrykcyjna weryfikacje danych pochodzących z formularza login-hasło,
- **(niezbędne)** przechowywanie hasła chronione funkcją hash, solą i pieprzem,
- **(niezbędne)** możliwość umieszczenia na serwerze haseł dostępnych prywatnie lub dla określonych użytkowników,
- **(niezbędne)** szyfrowanie symetryczne przechowywanych haseł.
- **(niezbędne)** zabezpieczenie transmisji poprzez wykorzystanie protokołu https,
- **(niezbędne)** możliwość zmiany hasła,
- **(niezbędne)** możliwość odzyskania dostępu w przypadku utraty hasła,
- dodatkowa kontrola spójności sesji (przeciw atakom XSRF),
- wielokrotne wykorzystanie funkcji hash, żeby wydłużyć ataki brute-force na hash,
- weryfikacja liczby nieudanych prób logowania,
- dodanie opóźnienia przy weryfikacji hasła w celu wydłużenia ataków zdalnych,
- sprawdzanie jakości hasła (jego entropii),
- informowanie użytkownika o nowych podłączeniach do jego konta.