



## SYLABUS PRZEDMIOTU

**Nazwa przedmiotu:** Projektowanie bezpiecznych architektur

**Kod przedmiotu:** SITA

**Kierunek / Profil:** Informatyka / praktyczny

**Tryb studiów:** stacjonarny

**Rok / Semestr:** 4 / 7

**Charakter:** obowiązkowy

**Odpowiedzialny:** mgr Adam Kassenberg

**Wersja z dnia:** 19.02.2026

### 1. Godziny zajęć i punkty ECTS

Wykłady	Ćwiczenia	Laboratoria	Z prowadzącym	Praca własna	Łącznie	ECTS
30 h	30 h	—	60 h	65 h	125 h	5

### 2. Forma zajęć

Forma zajęć	Sposób zaliczenia
Laboratorium	Zaliczenie z oceną
Wykład	Egzamin

### 3. Cel dydaktyczny

Ten kurs ma cel dostarczenia studentom kompleksowej wiedzy o bezpieczeństwie systemów informatycznych, takie jak poufność, integralność i dostępność danych. Nauczą się, jak projektować systemy infrastruktury informatyczne, które są odporne na różnorodne zagrożenia i ataki. Celem kursu będzie również zrozumienie mechanizmów zarządzania tożsamością i kontrolą dostępu, w tym wdrażanie polityk zabezpieczeń. Kurs obejmuje techniki i najlepsze praktyki

bezpiecznego programowania, aby zapobiegać wprowadzaniu luk bezpieczeństwa w fazie rozwoju oprogramowania. Studenci będą uczyć się, jak reagować na incydenty bezpieczeństwa, w tym jak przeprowadzić审计 i raportować naruszenia.

## 4. Treści programowe

---

1. 1. Wybór organizacji
2. Analiza różnych typów organizacji.
3. Przykłady typowych struktur organizacyjnych.
4. 2. Definicja celów biznesowych
5. Jak określić cele biznesowe organizacji.
6. Rola celów biznesowych w projektowaniu architektur IT.
7. Ćwiczenia praktyczne: Definiowanie celów biznesowych dla wybranej organizacji.
8. 3. Utworzenie diagramu przepływów DFD
9. Wprowadzenie do diagramów przepływów danych (DFD).
10. Tworzenie DFD dla wybranej organizacji.
11. Ćwiczenia praktyczne: Rysowanie DFD na podstawie zebranych danych.
12. 4. Procesowy opis organizacji wg ISO 9001
13. Wprowadzenie do standardu ISO 9001.
14. Opis procesów organizacyjnych zgodnie z ISO 9001.
15. Ćwiczenia praktyczne: Tworzenie procesowego opisu organizacji.
16. 5. Utworzenie architektury IT
17. Zasady projektowania architektur IT.
18. Elementy składowe architektury IT.
19. Ćwiczenia praktyczne: Projektowanie architektury IT dla wybranej organizacji.
20. 6. Koncepcja komunikacji, sprzętu oraz oprogramowania
21. Planowanie komunikacji w organizacji.
22. Wybór sprzętu i oprogramowania.
23. Ćwiczenia praktyczne: Opracowanie koncepcji komunikacji, sprzętu i oprogramowania.
24. 7. Klasyfikacja informacji, struktura zarządzania
25. Klasyfikacja informacji w organizacji.
26. Tworzenie struktury zarządzania informacją.
27. Ćwiczenia praktyczne: Klasyfikacja informacji i tworzenie struktury zarządzania.
28. 8. Identyfikacja zasobów organizacji
29. Identyfikacja i katalogowanie zasobów.
30. Zarządzanie zasobami organizacji.
31. Ćwiczenia praktyczne: Tworzenie katalogu zasobów organizacji.
32. 9. Utworzenie deklaracji stosowania
33. Definicja i znaczenie deklaracji stosowania.
34. Tworzenie deklaracji dla wybranej organizacji.
35. 10. Identyfikacja zagrożeń dla architektury

36. Typowe zagrożenia dla architektur IT.
37. Metody identyfikacji zagrożeń.
38. Ćwiczenia praktyczne: Analiza przypadków i identyfikacja zagrożeń.
39. 11. Identyfikacja podatności dla zagrożeń
40. Definicja podatności.
41. Metody identyfikacji podatności w systemach IT.
42. Ćwiczenia praktyczne: Identyfikacja podatności w wybranej architekturze IT.
43. 12. Szacowanie ryzyka dla wyodrębnionych zagrożeń
44. Metody szacowania ryzyka.
45. Przykłady zastosowania metod szacowania ryzyka.
46. Ćwiczenia praktyczne: Szacowanie ryzyka dla wybranych zagrożeń.
47. 13. Tworzenie rekomendacji po analizie ryzyka
48. Jak tworzyć rekomendacje bezpieczeństwa.
49. Przykłady rekomendacji po analizie ryzyka.
50. Ćwiczenia praktyczne: Tworzenie rekomendacji dla wybranej architektury IT.

## 5. Efekty kształcenia

---

### Wiedza

- Student zna i rozumie pojęcia takie jak poufność, integralność, dostępność danych. Student zna i rozumie zasady tworzenia bezpiecznych systemów infrastruktury informatycznej. Student zna i rozumie najlepsze praktyki kodowania, które minimalizują ryzyko wprowadzenia luk w bezpieczeństwie. Student zna i rozumie techniki zabezpieczania danych takie jak szyfrowanie, backup i monitoring. Student zna i rozumie międzynarodowe standardy bezpieczeństwa IT, takie jak ISO/IEC 27001 czy GDPR.

### Umiejętności

- Student potrafi identyfikować zagrożenia i oceniać ryzyko związane z różnymi systemami IT. Student potrafi zaimplementować mechanizmy kontroli dostępu i zarządzania tożsamością. Student potrafi zarządzać incydentami cyberbezpieczeństwa, analizować naruszenia i raportować.

## 6. Kryteria oceny

---

- Studium przypadków
- Kryteria oceny
- Kolokwium pisemne.
- Skala ocen:
- Poniżej 50% - ndst
- Od 50% - dst
- Od 60% - dst+

- Od 70% - db
- Od 80% - db+
- Od 90% - bdb
- Skala ocen:
- Poniżej 50% - ndst
- Od 50% - dst
- Od 60% - dst+
- Od 70% - db
- Od 80% - db+
- Od 90% - bdb

## 7. Metody dydaktyczne

---

Wykład, laboratoria, praca własna studenta.

## 8. Literatura

---

### Podstawowa:

- "Security Engineering: A Guide to Building Dependable Distributed Systems", Ross Anderson
- "Threat Modeling: Designing for Security", Adam Shostack
- "Cross Site Scripting (XSS) Attacks: Understanding and Preventing XSS Attacks", Robert Hansen

### Uzupełniająca:

- "Black Hat Python: Python Programming for Hackers and Pentesters" autorstwa Justin Seitz
- "SQL Injection Attacks and Defense" autorstwa Justin Clarke