W imieniu grupy
„KSZTAŁCENIE-ZAWODOWE.PL grupa dla nauczycieli zawodu i nie tylko”
skupiających nauczycieli i dyrektorów szkół kształcących w zawodach

Joanna Ksieniewicz
kontakt:
joanna.ksieniewicz@kwalifikacje-zawodowe.pl
Wyrażam zgodę na użycie moich danych osobowych i kontaktowych do celów konsultacji społecznych i opublikowanie niniejszego pisma w RCL.

Warszawa, 31.03.2026

Szanowni Państwo,
W ramach konsultacji społecznych dotyczących projektu rozporządzenia Ministra Edukacji zmieniające rozporządzenie w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkolnictwa branżowego poniżej przesyłamy zebrane uwagi dotyczące zawodu **technik cyberbezpieczeństwa**.

Uwagi były zbierane wśród: nauczycieli i dyrektorów, ochotników z różnych szkół (technika, szkoły policealne) kształcących w zawodach branży informatycznej z całej Polski uczestników nieformalnej grupy nauczycieli teoretycznych przedmiotów zawodowych i praktycznej nauki zawodu (w skrócie nauczycieli kształcenia zawodowego) oraz dyrektorów techników, szkół policealnych, centrów kształcenia praktycznego skupionych wokół portalu KSZTAŁCENIE-ZAWODOWE.PL w grupie „KSZTAŁCENIE-ZAWODOWE.PL grupa dla nauczycieli zawodu i nie tylko” (4368 członków na dzień 24.03.2026). Uwagi były zbierane między 16.03.2026 a 27.03.2026 zarówno na spotkaniach, które odbywały się online na platformie Google Meet oraz poprzez zgłoszenie pisemnych szczegółowych uwag. Spotkania poprowadziła i informacje zebrała Joanna Ksieniewicz - nauczyciel kształcenia zawodowego w zawodach branży INF i ELE, ekspert kształcenia zawodowego, od 2014 roku zajmuje się monitorowaniem projektów informatycznych w firmie zajmującej się sieciami i cyberbezpieczeństwem.

Technik cyberbezpieczeństwa - uwagi do projektu podstawy programowej kształcenia branżowego zebrane w ramach konsultacji społecznych

1. Nazwy kwalifikacji

Użyte nazwy kwalifikacji w projekcie podstawy programowej w zawodzie technik cyberbezpieczeństwa zawierają rzeczowniki odczasownikowe, które nie są na właściwym poziomie PRK - poziom PRK został przypisany w ramach innego rozporządzenia - rozporządzenia Ministra Edukacji zmieniającego rozporządzenie w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkoły rolnictwa branżowego (projekt z dnia 30.01.2026). Nazwy w projekcie podstawy programowej są również rozbieżne ze wspomnianym projektem rozporządzenia zmieniającym rozporządzenie w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkoły rolnictwa branżowego

Użyta nazwa kwalifikacji w projekcie podstawy programowej w zawodzie technik cyberbezpieczeństwa	**Użyta nazwa kwalifikacji w projekcie rozporządzenia zmieniającego w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkoły rolnictwa branżowego i odpowiadający poziom PRK**
INF.11. Zarządzanie bezpieczeństwem systemów i sieci komputerowych	INF.11. Bezpieczeństwo systemów i sieci komputerowych - PRK 4
INF.12. Programowanie bezpiecznych aplikacji webowych	INF.12 Bezpieczeństwo aplikacji webowych - PRK 5

Zgodnie z Rozporządzeniem Ministra Edukacji z dnia 20 marca 2026 r. zmieniające rozporządzenie w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkolnictwa branżowego (Dz.U.2026 poz. 404), które ma wejść w życie 01.09.2026, zostały wprowadzone natomiast nazwy takie same jak w projekcie podstawy programowej w zawodzie technik cyberbezpieczeństwa:

- INF.11. Zarządzanie bezpieczeństwem systemów i sieci komputerowych - przypisany do tego poziom PRK 4,
- INF.12. Programowanie bezpiecznych aplikacji webowych - przypisany do tego poziom PRK 5.

Po analizie szczegółowej zawartości kwalifikacji: pogrupowanych jednostek efektów kształcenia, przypisanych efektów kształcenia oraz kryterium weryfikacji można

stwierdzić, że nazwy kwalifikacji nie odpowiadają ich rzeczywistej zawartości merytorycznej oraz przypisanym im poziomom Polskiej Ramy Kwalifikacji (PRK).

W nazwie kwalifikacji INF.11 słowo „zarządzanie” na poziomie 4 PRK jest mylące. Zgodnie z deskryptorami PRK, poziom 4 dotyczy wykonywania zadań wymagających umiejętności rozwiązywania problemów w znanych kontekstach. Tymczasem zakres INF.11 obejmuje nie tylko administrację, ale przede wszystkim operacyjne utwardzanie infrastruktury i reagowanie na incydenty. Nazwa sugeruje rolę menedżerską („zarządzanie”), podczas gdy efekty kształcenia dotyczą fundamentów technicznych i operacyjnych niezbędnych do utrzymania ciągłości działania systemów i sieci. W tym przypadku nowa nazwa mogłaby brzmieć „INF.11. Zabezpieczanie systemów i sieci komputerowych”.

Kwalifikacja INF.12 określona na poziomie PRK 5 (poziom operacyjny/specjalistyczny) wymaga zdolności do samodzielnego projektowania rozwiązań i analizy podatności w zmiennych warunkach. Obecna nazwa redukuje tę kwalifikację do czynności czysto wytwórczych („programowanie”). Zawartość INF.12 obejmuje szeroki wachlarz testów penetracyjnych, weryfikacji bezpieczeństwa architektury oraz bezpieczeństwa chmurowego. Programowanie jest tu jedynie narzędziem do realizacji nadrzędnego celu, jakim jest inżynieria bezpieczeństwa aplikacji. W tym przypadku nowa nazwa kwalifikacji powinna określać raczej czynności w zakresie „Badania i zapewniania bezpieczeństwa aplikacji”. Co do rodzaju aplikacji - nazwa ta jest **zbyt wąska** i nieadekwatna do współczesnych standardów rynkowych oraz potrzeb kształcenia technika cyberbezpieczeństwa, bo pomija aplikacje desktopowe oraz aplikacje mobilne. W całej podstawie programowej w zakresie kwalifikacji INF.12 termin „aplikacje mobilne” pojawia się jedynie w sekcji „Miejsce realizacji praktyk zawodowych”. W efektach kształcenia i kryteriach weryfikacji nie ma ani jednego zapisu dotyczącego specyfiki zabezpieczania systemów Android/iOS, co przy dzisiejszym modelu pracy (BYOD – Bring Your Own Device) jest krytycznym brakiem. Podobnie jak w przypadku mobilnych, aplikacje desktopowe są wymienione jako potencjalne miejsce pracy, ale uczeń nie nabywa kompetencji w zakresie ich bezpiecznego tworzenia czy testowania. Podstawa skupia się wyłącznie na protokołach HTTP/HTTPS i architekturze klient-serwer w przeglądarce.
Poziom 5 PRK wymaga od absolwenta umiejętności projektowania i weryfikacji zabezpieczeń w szerokim spektrum środowisk. Ograniczenie kwalifikacji tylko do „aplikacji webowych” sugeruje, że technik cyberbezpieczeństwa nie posiada kompetencji do analizy bezpieczeństwa oprogramowania zainstalowanego lokalnie na stacjach roboczych czy smartfonach. Ponadto współczesne ataki często wykorzystują aplikacje desktopowe i mobilne jako wejście do infrastruktury (np. poprzez złośliwe aplikacje komunikacyjne), a podstawa programowa pozostawia te obszary niezagospodarowane.

2. Wyposażenie szkoły niezbędne do realizacji kształcenia

2.1. Wyposażenie niezbędne w ramach pierwszej kwalifikacji INF.11.

W ramach wyposażenia Pracowni bezpieczeństwa systemów i sieci komputerowych zawiera zapisy:

1) dwunasty tiret: “− zestaw zarządzalnych fizycznych przełączników (warstwa 2 i 3) oraz routerów (jeden zestaw na dwóch uczniów)”

   Uwaga: Zapis ten nie precyzuje konkretnie czym jest zestaw (ile i jakie urządzenia wchodzą w skład zestawu) - zapis wymaga doprecyzowania

   Propozycja zmian:

Obecny zapis	Uwagi do zmian
zestaw zarządzalnych fizycznych przełączników (warstwa 2 i 3) oraz routerów (jeden zestaw na dwóch uczniów	zapis wymaga doprecyzowania co wchodzi w skład zestawu w zakresie minimum

2) piętnasty tiret: “− serwer lub stacje robocze o zasobach pamięci RAM umożliwiających uruchomienie środowiska wielosystemowego”

   Uwagi:

   a) błąd stylistyczny i logiczny - fraza "umożliwiających uruchomienie" nie definiuje ona konkretnego parametru technicznego, lecz oczekiwany rezultat, który jest zmienny w zależności od wersji użytego oprogramowania.
   b) brak zdefiniowanego standardu “środowiska wielosystemowego” - “środowisko wielosystemowe" w kontekście egzaminacyjnym może oznaczać różne wymagania: czy to dwie lekkie dystrybucje Linuxa, czy może Windows Server 2022 uruchomiony jednocześnie z dwiema stacjami Windows 11 i maszyną do testów? Różnica w zapotrzebowaniu na RAM między tymi scenariuszami wynosi od 8 GB do nawet 64 GB.
   c) niespójność z wymaganiami INF.12 - w dalszej części dokumentu (dla kwalifikacji INF.12) autor projektu podstawy programowej był już bardziej konkretny, wpisując “stacje robocze o zasobach sprzętowych (min. 16 GB RAM) umożliwiających równoległe uruchamianie kontenerów aplikacyjnych i narzędzi testowych”. Brak analogicznego doprecyzowania (minimum 16 GB lub 32 GB) w INF.11 tworzy niebezpieczną lukę interpretacyjną.

Propozycja zmian

Obecny zapis	Propozycja zmiany
serwer lub stacje robocze o zasobach pamięci RAM umożliwiających uruchomienie środowiska wielosystemowego	stacje robocze wyposażone w minimum 32 GB pamięci RAM, umożliwiające jednoczesną i płynną pracę co najmniej trzech systemów operacyjnych w środowisku zwirtualizowanym

3) szesnasty tiret “− zewnętrzne nośniki danych do kopii zapasowych”
    a) Zapis - jest to kolejny przykład sformułowania zbyt ogólnego, które w realiach nowoczesnego cyberbezpieczeństwa (szczególnie dla poziomu PRK 4/5) jest technologicznie niewystarczające
    b) Nieokreślona definicja nośnika - w obecnym brzmieniu zapis ten sugeruje proste rozwiązania typu pendrive lub przenośny dysk USB. W kontekście efektów kształcenia dla kwalifikacji INF.11, które obejmują m.in.:
        - wdrażanie zasady kopii zapasowych
        - administrowanie systemami zarządzania bazą danych (SZBD) i wykonywanie pełnych oraz przyrostowych kopii.
        - zarządzanie kopiami zapasowymi w środowiskach wirtualnych i kontenerowych,

       Stosowanie wyłącznie prostych nośników zewnętrznych uniemożliwia przećwiczenie automatyzacji, retencji danych czy bezpiecznego składowania sieciowego (np. odpornego na ransomware).
    c) Brak spójności z dydaktyką "utwardzania" - projekt podstawy programowej kładzie duży nacisk na utwardzanie systemów i izolację. Fizyczne nośniki USB często stanowią wektor ataku (np. złośliwe oprogramowanie na kontrolerze USB), co uczeń ma potrafić rozpoznawać. Opieranie strategii backupu wyłącznie na takich narzędziach jest dydaktycznie niespójne z nauką o bezpiecznej infrastrukturze.

    Propozycja zmian
    Aby przygotować ucznia do realiów rynkowych, wyposażenie powinno uwzględniać urządzenia pozwalające na realizację backupu sieciowego i logiczną izolację danych.

Obecny zapis	Propozycja zmiany
zewnętrzne nośniki danych do kopii	sieciowa pamięć masowa (NAS) z

zapasowych	obsługą protokołów bezpiecznej transmisji oraz dedykowane nośniki zewnętrzne umożliwiające realizację strategii kopii zapasowych 3-2-1

4) siedemnasty tiret “− testery okablowania i zestaw narzędzi monterskich”

Uwaga:
Zapis ten nie precyzuje konkretnie czym jest zestaw i jakiego rodzaju testery okablowania powinny być w zestawie minimum - zapis wymaga doprecyzowania

Propozycja zmian:

Obecny zapis	Uwagi do zmian
testery okablowania i zestaw narzędzi monterskich”	zapis wymaga doprecyzowania co wchodzi w skład zestawu w zakresie minimum i jakiego rodzaju testery okablowania

5) dziewiętnasty tiret “− platformy chmurowej (publicznej lub hybrydowej) umożliwiającej praktyczne wykorzystanie modeli serwisowych (IaaS, PaaS) oraz zarządzanie bezpieczeństwem w chmurze, w tym:
− dostęp do konsoli administracyjnej umożliwiającej samodzielne tworzenie wirtualnych sieci prywatnych, maszyn wirtualnych Linux oraz grup bezpieczeństwa,
− środowiska pozwalającego na wdrażanie i orkiestrację kontenerów wraz z prywatnym rejestrem obrazów,
− narzędzi do nauki kreowania ról, uprawnień i wdrażania uwierzytelniania wieloskładnikowego dla użytkowników i usług,
− dostępu do usług bezpiecznego przechowywania kluczy,
− chmurowych systemów zbierania logów i analizy zdarzeń, pozwalających na wykrywanie incydentów w czasie rzeczywistym.”
Uwagi:
Zapis dotyczący **platformy chmurowej** wykazuje, że jest to jeden z najbardziej wymagających i kosztownych punktów w całym projekcie podstawy programowej. Ten zapis nakłada na szkołę obowiązek stania się "klientem biznesowym" dostawców usług chmurowych (np. AWS, Azure, Google Cloud), co rodzi szereg problemów formalno-prawnych i finansowych.

a) analiza merytoryczna wymagań - projekt podstawy wymaga dostępu do pełnego spektrum usług nowoczesnej chmury w zakresie:
    - IaaS i PaaS: Szkoła musi zapewnić środowisko do tworzenia sieci wirtualnych (VPC), maszyn wirtualnych (VM) oraz grup bezpieczeństwa (firewalli chmurowych)
    - Orkiestracja kontenerów: Wymagany jest dostęp do usług typu Kubernetes (AKS/EKS) lub zarządzanych kontenerów wraz z prywatnym rejestrem obrazów
    - Identity and Access Management (IAM): Nauka ról, uprawnień i obowiązkowego MFA (uwierzytelniania wieloskładnikowego)
    - Zarządzanie sekretami: Dostęp do usług typu Key Vault / KMS
    - Security Operations (SecOps): Systemy zbierania logów i analizy zdarzeń w czasie rzeczywistym
b) bariery do realizacji zapisu wyposażenia:
    - Bariera prawno-administracyjna - rejestracja w chmurach publicznych (AWS/Azure) zazwyczaj wymaga karty płatniczej przypisanej do konta. Jednostki budżetowe (szkoły) mają ogromne trudności z rozliczaniem płatności kartowych oraz zmiennych faktur w walutach obcych (USD/EUR)
    - Bariera kompetencyjna - zarządzanie taką platformą dla 30 uczniów wymaga od nauczyciela kompetencji na poziomie certyfikowanego architekta chmury, a projekt podstawy programowej nie precyzuje, kto ma administrować tą "szkolną chmurą"
    - Bariera bezpieczeństwa - samodzielne tworzenie przez uczniów wirtualnych sieci i maszyn z dostępem do internetu bez ścisłego nadzoru może narazić szkołę na ogromne koszty (np. przypadkowe uruchomienie drogich instancji) lub ataki typu cryptojacking.
c) analiza kosztów
   W chmurze publicznej nie płaci się za "zakup", lecz za "użycie" (Pay-as-you-go).

Usługa	Szacowany koszt na ucznia / m-c	Uwagi
Maszyny wirtualne (IaaS)	40 - 100 PLN	Przy założeniu pracy przez kilka godzin tygodniowo
Orkiestracja (Kubesnetes)	150 - 400 PLN	Usługi zarządzane (np. AKS) mają wysoką opłatę stałą za klaster

Logi i Analiza (SIEM)	50 - 150 PLN	Płatne za ilość przetworzonych danych (GB)
SUMA dla pracowni na 15 osób	3600 - 9000 PLN/mc	Koszt stały, który szkoła musi zabezpieczyć w budżecie co miesiąc

Alternatywa - chmura hybrydowa:
Zapis dopuszcza "platformę hybrydową". Oznacza to, że szkoła mogłaby zainstalować własny serwer z oprogramowaniem typu OpenStack lub Proxmox. Zaletą tego rozwiązanie jest brak miesięcznych opłat dla firm IT, pełna kontrola nad infrastrukturą. Wadą natomiast jest bardzo wysoki koszt początkowy serwera (min. 40 000 – 60 000 PLN) oraz konieczność posiadania eksperta na miejscu do konfiguracji.

Propozycja zmian:
Wymóg korzystania z komercyjnych platform chmurowych bez zapewnienia systemowego finansowania i gotowych scenariuszy konfiguracji narazi szkoły na niekontrolowane koszty oraz problemy z rozliczaniem usług o charakterze zmiennym. Należy doprecyzować, czy państwo zapewni dostęp do dedykowanej infrastruktury chmurowej dla zawodu technik cyberbezpieczeństwa lub zapewni centralny system grantowy czy dedykowaną platformę edukacyjną dla szkół.

2.2. Braki sprzętowe w wyposażeniu niezbędne w ramach pierwszej kwalifikacji INF.11.

W tej kwalifikacji braki są najbardziej dotkliwe, ponieważ to tutaj uczeń ma nabyć fundamenty techniczne zarządzania dostępem i odporności danych.

1) **Klucze sprzętowe (U2F/FIDO2)**: W kryteriach weryfikacji zapisano, że uczeń „konfiguruje uwierzytelnianie wieloskładnikowe (np. wykorzystanie [...] kluczy sprzętowych Universal 2nd Factor (U2F))”. Tymczasem w spisie wyposażenia pracowni dla INF.11 nie ma ani słowa o posiadaniu takich kluczy. Powinno się zapewnić posiadanie takich kluczy (zestawu kluczy) w odpowiedniej liczbie na stanowisko.
2) **Nośniki danych i Backup**: Uczeń ma wdrażać „zasadę 3-2-1”. Wyposażenie pracowni przewiduje jedynie „zewnętrzne nośniki danych do kopii zapasowych”. Bez serwera NAS lub dedykowanego rozwiązania sieciowego realizacja tej zasady (szczególnie w aspekcie „2 różnych nośników”) jest w warunkach szkolnych utrudniona i mało profesjonalna.
3) **Pamięć RAM:** Zapis o „zasobach RAM umożliwiających uruchomienie środowiska wielosystemowego” jest nieprecyzyjny, co przy wymagających

systemach SIEM i wirtualizacji może uniemożliwić realizację zajęć na sprzęcie o zbyt niskich parametrach (np. 8 GB)

2.3. Wyposażenie niezbędne w ramach drugiej kwalifikacji INF.12.

W ramach wyposażenia wprowadzono następujące zapisy

1) pierwszy i drugi tiret:
   "− stanowisko komputerowe dla nauczyciela, z urządzeniem wielofunkcyjnym oraz z projektorem multimedialnym lub tablicą interaktywną,
   − stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia), wszystkie komputery podłączone do sieci lokalnej z dostępem do internetu, do urządzeń wielofunkcyjnych, pakiet programów biurowych, program do wspomagania projektowania i wykonywania diagramów"

   Uwagi:
   W powyższych zapisach stosowana jest raz forma w liczbie pojedynczej, raz w liczbie mnogiej "urządzenia wielofunkcyjnego". Zapis o „urządzeniu wielofunkcyjnym" w pracowniach specjalistycznych (INF.11 i INF.12) jest sformułowaniem standardowym dla podstaw programowych, ale w kontekście cyberbezpieczeństwa nabiera on specyficznego znaczenia operacyjnego i ryzyka, a za nim kryją się problemy związane z realizacją zapisów
   a) klasyczna definicja "urządzenia wielofunkcyjnego" - w najprostszym ujęciu urządzenie wielofunkcyjne to sprzęt łączący funkcje drukarki, skanera i kopiarki. W warunkach szkolnych, w kontekście realizacji kształcenia w zawodzie technik cyberbezpieczeństwa" będzie służyć do drukowania dokumentacji technicznej, raportów z testów podatności oraz schematów sieci, a także do skanowania protokołów przekazania sprzętu lub podpisanych oświadczeń o zachowaniu poufności (NDA), które są elementem nauki etyki i procedur
   b) "urządzenie wielofunkcyjne" w interpretacji rozszerzonej w kontekście realizacji zadań zawodowych dla technika cyberbezpieczeństwa - urządzenie to nie jest tylko biurowym dodatkiem, ale pełnoprawnym elementem infrastruktury IT, który podlega tym samym rygorom, co serwer czy stacja robocza
      - Urządzenie jako wektor ataku - nowoczesne urządzenia wielofunkcyjne posiadają własne systemy operacyjne, pamięć masową i interfejsy sieciowe. W ramach kształcenia uczeń powinien uczyć się ich utwardzania (zmiana haseł domyślnych, wyłączanie podatnych protokołów jak SNMP v1/v2 czy FTP)
      - Audyt i logowanie - Urządzenie powinno umożliwiać integrację z systemem zbierania logów (np. syslog), co pozwala na monitorowanie, kto i kiedy skanował lub drukował wrażliwe dane

- Bezpieczeństwo danych - skaner musi wspierać bezpieczne protokoły przesyłania plików (np. SFTP zamiast FTP) oraz szyfrowanie komunikacji (TLS/SSL).

c) koszty i wymagania dla szkoły - jeśli szkoła zakupi najtańsze urządzenie „domowe”, nie będzie w stanie zrealizować części efektów kształcenia dotyczących administracji i bezpieczeństwa sieci - poniżej tabela ilustrująca różnice w cenach i możliwościach dydaktycznych

Klasa urządzenia	Szacunkowy koszt	Możliwości dydaktyczne
Podstawowa (Inkjet/Laser home)	800 – 1 500 PLN	Tylko druk i skan. Brak funkcji zarządzania bezpieczeństwem sieciowym.
Biznesowa (Workgroup MFP)	3 500 – 7 000 PLN	Obsługa LDAP/Active Directory, certyfikatów 802.1X, szyfrowanego skanowania i logowania zdarzeń.

d) ryzyko niespójności w dokumentacji - w pierwszym tirecie wymieniono “stanowisko komputerowe dla nauczyciela z urządzeniem wielofunkcyjnym” (liczba pojedyncza), a w wyposażeniu dla uczniów wymieniono “wszystkie komputery podłączone do sieci lokalnej z dostępem do (...) urządzeń wielofunkcyjnych” (liczba mnoga)

Propozycja zmian:

Należy doprecyzować, czy szkoła ma obowiązek zapewnić jedno centralne, zaawansowane urządzenie sieciowe (klasy biznesowej), czy też mniejsze urządzenia na każdym stanowisku (co byłoby ekonomicznie nieuzasadnione). Można rozważyć zapis „sieciowe urządzenie wielofunkcyjne klasy biznesowej z obsługą protokołów bezpiecznej transmisji i autoryzacji”.

2) trzeci tiret “− oprogramowanie do wirtualizacji systemów operacyjnych, systemy operacyjne z rodziny Linux oraz Windows, środowisko do konteneryzacji (Docker),”
siódmy tiret “− oprogramowanie do konteneryzacji (Docker) wykorzystywane do budowania i izolacji bezpiecznych środowisk uruchomieniowych aplikacji oraz baz danych wraz z oprogramowaniem umożliwiającym zarządzaniem kontenerami i skanowaniem ich podatności,”

Uwagi:

Wskazywanie konkretnej technologii wprowadza niespójność w zapisie projektu podstawy programowej, gdzie zazwyczaj podstawy programowe zachowują neutralność w zakresie używanej technologii. Zaletą Dockera jest fakt, że standardem rynkowym, więc uczeń zyskuje realną kompetencję, natomiast wadą jest to, że konteneryzacja na Windowsie (Docker Desktop) wymaga włączenia mechanizmów WSL2 lub Hyper-V, co drastycznie podnosi wymagania co do procesora i pamięci RAM (kolejny argument za min. 32 GB RAM.

Propozycja zmian:
Nie należy narzucać jednej technologii. Należy doprecyzować, że systemy z rodziny Windows muszą być dostarczone w wersji umożliwiającej zarządzanie politykami grup (GPO) oraz mechanizmami izolacji poświadczeń (min. wersja Professional lub Enterprise), aby umożliwić realizację efektów kształcenia zawartych w jednostkach INF.11.6 oraz INF.12.4.

3) jedenasty tiret "− platformy chmurowej (publicznej lub hybrydowej) umożliwiającej praktyczne wykorzystanie modeli serwisowych (IaaS, PaaS) oraz zarządzanie bezpieczeństwem w chmurze, w tym:
− konsoli administracyjnej umożliwiającej samodzielne tworzenie wirtualnych sieci prywatnych, maszyn wirtualnych Linux oraz grup bezpieczeństwa,
− środowiska pozwalającego na wdrażanie i orkiestrację kontenerów wraz z prywatnym rejestrem obrazów,
− narzędzi do nauki kreowania ról, uprawnień i wdrażania uwierzytelniania wieloskładnikowego dla użytkowników i usług,
− usług bezpiecznego przechowywania kluczy,
− chmurowych systemów zbierania logów i analizy zdarzeń, pozwalających na wykrywanie incydentów w czasie rzeczywistym."

Uwagi - analogiczne jak zgłoszone w przypadku kwalifikacji INF.11 (ten sam zakres) - patrz uwaga 2.2.1.5).

2.4. Braki sprzętowe w wyposażeniu niezbędne w ramach pierwszej kwalifikacji INF.12.

W tej kwalifikacji występują niespójności w zakresie:

1) **Klucze sprzętowe**: Podobnie jak w INF.11, uczeń ma umieć wykorzystywać „klucze sprzętowe Universal 2nd Factor (U2F) dla personelu z uprawnieniami administracyjnymi". W wyposażeniu pracowni dla INF.12 również ich nie uwzględniono.
2) **Pamięć RAM**: Tutaj autor projektu podstawy programowej w zawodzie technik cyberbezpieczeństwa był bardziej rzetelny, określając minimum na poziomie 16 GB RAM. Jest to jednak wartość graniczna (minimum), która

przy równoległym uruchamianiu kontenerów Docker, serwerów baz danych i narzędzi takich jak Burp Suite/ZAP może okazać się niewystarczająca do płynnej nauki.

3) **Brak fizycznych zapór ogniowych:** O ile w INF.11 przewidziano „sprzętową zaporę sieciową (firewall)" , o tyle w INF.12 uczeń ma konfigurować „chmurowe zapory sieciowe" i „reverse proxy z modułem WAF", ale w wyposażeniu pracowni brakuje fizycznego odpowiednika do testów lokalnych.

3. Zakres realizowanych zadań

3.1. Realizacja zadań zawodowych w kontekście potwierdzania ich na egzaminie praktycznym

Przy obecnym brzmieniu projektu podstawy programowej, autorzy zadań egzaminacyjnych mogą napotkać trudności w standaryzacji i obiektywnej ocenie umiejętności uczniów. Zakres problemów przy potwierdzaniu kwalifikacji egzaminem praktycznym w podziale na kwalifikacje:

1) Kwalifikacja INF.11
   Kwalifikacja ta skupia się na administracji i utwardzaniu infrastruktury. Na egzamin praktyczny będą mieć wpływ następujące zakresy:
   a) Czasochłonność wirtualizacji jako wyzwanie egzaminacyjne - egzamin praktyczny w zakresie "użytkowania, konfiguracji i utwardzania systemów" wymaga uruchomienia co najmniej dwóch maszyn wirtualnych (np. serwer i stacja robocza). Przy obecnych brakach w precyzji opisu w projekcie podstawy programowej w zakresie pamięci RAM dla INF.11, proces bootowania i konfiguracji systemów może skonsumować znaczną część czasu egzaminacyjnego, uniemożliwiając uczniowi wykonanie zadań merytorycznych.
   b) Problem z "etycznym hackingiem" - wymaga rozróżniania etycznego hackingu od przestępstwa oraz testowania bezpieczeństwa sieci. Realizacja tego na egzaminie praktycznym wymaga stworzenia całkowicie odizolowanego środowiska laboratoryjnego (sandbox), aby działania egzaminacyjne nie wpłynęły na stabilność sieci szkolnej.
   c) Brak standaryzacji narzędzi wykorzystujących sztuczną inteligencję - Kryteria weryfikacji wskazują na wykorzystanie narzędzi AI do weryfikacji kodu. Na egzaminie praktycznym dostęp do zewnętrznych modeli LLM (np. ChatGPT) nie będzie możliwy ze względów bezpieczeństwa i samodzielności pracy. Bez lokalnych instancji AI, ten punkt programu jest nieegzaminowalny w obecnej formie.
2) Kwalifikacja INF.12
   Ta kwalifikacja jest znacznie bardziej złożona. Na egzamin praktyczny będą mieć wpływ następujące zakresy:
   a) Dualizm technologiczny (Java vs C#) - dopuszczenie dwóch różnych języków programowania to ogromne wyzwanie dla Komisji

Egzaminacyjnych. Przygotowanie dwóch równoważnych arkuszy egzaminacyjnych, które w takim samym stopniu sprawdzają "bezpieczeństwo implementacji", jest niemal niemożliwe, co może prowadzić do nierównego traktowania zdających.

b) Egzamin praktyczny w zakresie testów penetracyjnych - zadanie polegające na "przeprowadzaniu testów penetracyjnych aplikacji webowych" jest z natury procesem kreatywnym i nieliniowym. Trudno ująć je w sztywny klucz oceniania (0/1), który dominuje w polskim systemie egzaminów zawodowych. Istnieje ryzyko, że egzamin skupi się na mechanicznych czynnościach, a nie na realnej analizie podatności albo stanie się egzaminem praktycznym wykonywanym wyłącznie w formie dokumentacji, co będzie z dużą niekorzyścią dla zdających uczniów.

c) Zależność od Chmury - wymóg wdrażania aplikacji w środowisku chmurowym stwarza problem techniczny podczas egzaminu: co jeśli w dniu egzaminu wystąpi awaria u dostawcy chmury (np. Azure/AWS) lub szkoła będzie miała problem z łączem internetowym? Egzamin zawodowy powinien być możliwy do przeprowadzenia w 100% lokalnie.

Autor/autorzy projektu podstawy programowej powinni na etapie pisania podstawy formułować zadania zawodowe w takiej formie, aby była możliwość ich realnego sprawdzenia egzaminem praktycznym.

3.2. Przypisanie jednostek efektów kształcenia do odpowiedniej kwalifikacji

W kontekście analizy przypisanych jednostek efektów kształcenia do danej kwalifikacji:

1) obecna jednostka efektów kształcenia “INF.11.4. Projektowanie i administracja relacyjnymi bazami danych” jest przypisana do kwalifikacji INF.11
   Uwaga:
   JEK “Projektowanie i administracja relacyjnymi bazami danych” powinna zostać przeniesiona do realizacji z kwalifikacji “sieciowej INF.11 do kwalifikacji “aplikacyjnej” INF.12. Zmiana ta ma głębokie uzasadnienie w architekturze współczesnych systemów IT.
   a) niespójność z profilem kwalifikacji INF.11
      - INF.11 (poziom PRK 4) ma przygotowywać do roli administratora systemów i sieci - administrator sieci zajmuje się głównie warstwą transportową i systemową (routery, przełączniki, utwardzanie OS), a nie strukturą logiczną bazy danych.
      - INF.12 (poziom PRK 5) dotyczy bezpieczeństwa aplikacji - współczesne ataki na aplikacje webowe (np. SQL Injection, o którym mowa w kryteriach INF.12) uderzają bezpośrednio w bazę danych i zrozumienie struktury SQL, normalizacji i

procedur składowanych jest niezbędne na poziomie programistycznym, a nie sieciowym.

b) dublowanie i luki w efektach kształcenia - w obecnym projekcie podstawy programowej występuje niedopuszczalna sytuacja
   - w INF.11.4 uczeń uczy się zaawansowanego projektowania, diagramów związków encji (ERD) i normalizacji
   - W INF.12.3 (Implementacja aplikacji) wymaga się od ucznia "implementacji warstwy dostępu do bazy danych", mimo że teoretyczne podstawy jej budowy zostały przypisane do innej, niższej kwalifikacji

   Dlatego też przeniesienie tej jednostki do INF.12 pozwoli na naturalne połączenie nauki programowania (Java/C#) z nauką baz danych (SQL).
c) przypisanie do właściwego poziomu PRK - projektowanie bezpiecznych i wydajnych struktur bazodanowych (w tym obsługa transakcji i audytowalność operacji) to zadanie o wysokim stopniu złożoności, które lepiej wpisuje się w poziom 5 PRK (INF.12) niż w poziom 4 (INF.11).

Propozycja zmian:

Obecna sytuacja	**Uwagi do zmian**
Jednostka efektów kształcenia “Projektowanie i administracja relacyjnymi bazami danych” jest przypisana do kwalifikacji INF.11 pod numerem INF.11.4	Przenieść całą obecną jednostkę kształcenia “Projektowanie i administracja relacyjnymi bazami danych” do kwalifikacji INF.12

Ta zmiana jest korzystna dla ucznia, ponieważ uczeń kończący INF.11 potrafi zabezpieczyć serwer, na którym stoi baza, natomiast uczeń kończący INF.12 potrafi zaprojektować bezpieczną aplikację, która z tą bazą się komunikuje. To jest podział zgodny ze standardami rynkowymi.

2) W konsekwencji pkt 1) analogicznie powinno się przenieść odpowiadające jednostce efektów kształcenia zadanie zawodowe z kwalifikacji INF.11 do INF.12

Propozycja zmian:

Obecna sytuacja	**Uwagi do zmian**
Zadanie zawodowe z kwalifikacji INF.11 “b) projektowania i administracji relacyjnymi bazami danych”	Przenieść zadanie zawodowe “b) projektowania i administracji relacyjnymi bazami danych” do kwalifikacji INF.12

4. Analiza szczegółowa jednostek efektów kształcenia w zakresie efektów kształcenia i kryteriów weryfikacji

4.1. Dotyczy INF.11.1. i INF.12.1 Bezpieczeństwo i higiena pracy

Obecne zapisy efektów w obszarze BHP skupiają się wyłącznie na “pracy biurowej” i “stanowiskach komputerowych”. W kontekście analizy efektów kształcenia oraz kryteriów weryfikacji nie uwzględniono specyfiki pracy w zawodzie technik cyberbezpieczeństwa:

1) Specyfika pracy w centrach danych (Data Center) - praca w serwerowniach jest pracą w trudnych warunkach - występują specyficzne zagrożenia: hałas o wysokim natężeniu, systemy gaszenia gazem (ryzyko uduszenia), praca przy szafach rack (ciężkie urządzenia) oraz zagrożenia elektryczne prądem stałym o dużym natężeniu.
   Propozycja zmian:
   Do kryteriów weryfikacji w punkcie dotyczącym zagrożeń warto dopisać: „identyfikuje zagrożenia specyficzne dla pracy w serwerowniach i centrach danych”

2) Higiena cyfrowa i ergonomia poznawcza - praca w cyberbezpieczeństwie (szczególnie w jednostkach SOC/Incident Response) wiąże się z ogromnym obciążeniem psychicznym, pracą w trybie 24/7 i tzw. „zmęczeniem alertami” (alert fatigue).
   Propozycja zmian:
   W zakresie ergonomii warto rozważyć zapis: „opisuje zasady higieny cyfrowej oraz zarządzania obciążeniem poznawczym w pracy”

3) Procedury w sytuacjach awaryjnych IT (Disaster Recovery) - w cyberbezpieczeństwie „bezpieczeństwo i higiena” to także ochrona danych przed fizycznym zniszczeniem (np. zalanie, pożar serwerowni).
   Propozycja zmian:
   W zakresie ochrony danych przed fizycznym zniszczeniem „stosuje procedury bezpieczeństwa fizycznego i ochrony mienia w sytuacji wystąpienia fizycznego incydentu w infrastrukturze IT”.

4) Niespójność w nazewnictwie efektów - INF.11.1 w 5. efekcie kształcenia pojawia się zapis: „organizuje stanowisko pracy zgodnie z wymaganiami ergonomii oraz przepisami dotyczącymi bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej i ochrony środowiska”. Jest to sformułowanie bardzo ogólne, powielane z innych zawodów, które nie uwzględnia, że „stanowiskiem pracy” technika jest często środowisko zwirtualizowane.
   Propozycja zmian:
   Wprowadzenie rozróżnienia na fizyczne stanowisko pracy oraz bezpieczeństwo pracy w środowiskach rozproszonych/zdalnych.

Podsumowanie propozycji zmian w zakresie zapisów BHP:
W jednostkach efektów kształcenia INF.11.1 oraz INF.12.1 należy uzupełnić kryteria weryfikacji o zagrożenia specyficzne dla branży cyberbezpieczeństwa, takie jak:

- praca w środowisku serwerowni (hałas, systemy gaszenia gazem),
- ergonomia poznawcza (alert fatigue),
- procedury bezpieczeństwa fizycznego infrastruktury krytycznej.

Dodatkowo, w INF.12.1 (PRK 5) kryteria powinny zostać rozszerzone o elementy planowania i nadzoru nad bezpieczeństwem pracy zespołu, aby odzwierciedlić wyższy poziom kwalifikacji (np. przygotowywanie instrukcji, prowadzenie instruktażu).

4.2. Dotyczy INF.11.2. i INF.12.2 Podstawy cyberbezpieczeństwa systemów IT

4.2.1. Analiza zapisów efektów kształcenia i kryteriów weryfikacji z zakresu podstaw cyberbezpieczeństwa systemów IT

1. Niespójność celów z kryteriami weryfikacji (Teoretyzacja)
Cele kształcenia dla zawodu technik cyberbezpieczeństwa zakładają przygotowanie do zadań takich jak „zarządzanie bezpieczeństwem", „konfiguracja i utwardzanie systemów" czy „testy penetracyjne". Tymczasem kryteria weryfikacji w jednostkach INF.11.2 i INF.12.2 są niemal całkowicie oparte na wiedzy deklaratywnej (teoretycznej):

- Dominacja czasowników pasywnych -w większości punktów uczeń ma jedynie: „wyjaśniać", „opisywać", „wymieniać", „rozróżniać" lub „wskazywać" .
- Brak operacjonalizacji - np. w zakresie kryptografii uczeń ma „wyjaśnić sposób działania" lub „wskazać standardy" , zamiast praktycznie zaimplementować szyfrowanie lub zweryfikować poprawność certyfikatu w konsoli systemowej.

2. Anachronizm i „buzzwordy" technologiczne
Zapisy wykazują brak zrozumienia dynamiki branży IT:

- Kryptografia postkwantowa (PQC) - wprowadzenie kryterium „wyjaśnia wpływ komputerów kwantowych na klasyczną kryptografię" na poziomie technikum jest przejawem przeładowania programowego treściami akademickimi kosztem rzetelnych podstaw. Na poziomie technika kluczowe jest sprawne zarządzanie infrastrukturą PKI, a nie teoretyczna analiza mechanizmów postkwantowych.
- Sztuczna inteligencja (AI) - kryterium dotyczące „wspomagania oceny wiarygodności przekazu z wykorzystaniem narzędzi AI" jest sformułowane zbyt ogólnikowo. Brakuje wskazania konkretnych grup narzędzi lub

metodologii, co przy tempie rozwoju AI (zmiany co kilka miesięcy) uczyni ten zapis nieużytecznym bez stałej aktualizacji programu.

3. Błędy dydaktyczne i brak gradacji trudności (PRK)
Analiza wykazuje chaos w przypisaniu poziomów trudności do kryteriów, co narusza zasady Polskiej Ramy Kwalifikacji:

- Mieszanie poziomów - w ramach jednej jednostki efektów (np. efekt 8: operacje kryptograficzne) obok siebie występują proste czynności, jak „oblicza skróty (hash)" , oraz niezwykle złożone procesy, jak „generuje certyfikat własnego CA (Certificate Authority)". Procesy te wymagają zupełnie innych nakładów czasu i poziomu abstrakcji, a zostały zrównane w strukturze dokumentu.
- Brak integracji z prawem - jednostka kończy się szerokim blokiem dotyczącym Krajowego Systemu Cyberbezpieczeństwa (KSC) i odpowiedzialności karnej. Dydaktycznie nauczanie aspektów prawnych w oderwaniu od technicznych procedur reagowania na incydenty (incident response) jest mało efektywne i prowadzi do wyuczenia się definicji, a nie zrozumienia obowiązków operatora usług kluczowych w praktyce.

4.2.2. Omyłki pisarskie

Efekt kształcenia	**Kryterium weryfikacji przed zmianą**	Kryterium weryfikacji **po zmianie**
2) charakteryzuje zagrożenia cyberbezpieczeństwa	3) opisuje działanie różnych typów złośliwego oprogramowania (np. wirus, spyware, ransomware, trojan, backdoor)	**4)** opisuje działanie różnych typów złośliwego oprogramowania (np. wirus, spyware, ransomware, trojan, backdoor)
2) charakteryzuje zagrożenia cyberbezpieczeństwa	4) wskazuje cele i potencjalne skutki ataków:	**5)** wskazuje cele i potencjalne skutki ataków:
2) charakteryzuje zagrożenia cyberbezpieczeństwa	5) opisuje proste scenariusze ataków:	**6)** opisuje proste scenariusze ataków:

4.2.3. Uwagi dotyczące wybranych efektów kształcenia

Efekt kształcenia	Kryterium weryfikacji	Uwagi
5) stosuje uwierzytelnianie wieloskładnikowe MFA (Multi-Factor Authentication)	3) podaje przykłady metod MFA (np. OTP (One-Time Password), Google Authenticator TOTP (Time-Based OTP), token, biometria, techniki behawioralne)	Podstawa programowa nie powinna wskazywać konkretnych rozwiązań komercyjnych
6) stosuje zasady kontroli dostępu	7) stosuje modele i mechanizmy kontroli dostępu (np. kontrola oparta na rolach (RBAC), kontrola oparta na atrybutach (ABAC), kontrola obowiązkowa (MAC), listy kontroli dostępu (ACL))	Brak wyjaśnienia pojęć. Należy na początek wprowadzić pojęcia, a następnie omówić mechanizmy i modele

Rekomendancje:

1) Zmiana czasowników operacyjnych - zastąpienie „opisuje/wyjaśnia” czasownikami takimi jak: „przeprowadza”, „konfiguruje”, „testuje”, „wykrywa”, „zabezpiecza”. Tylko to pozwoli na przeprowadzenie egzaminu w formie wykonawczej.
2) Urealnienie kryptografii - usunięcie teoretycznych rozważań o komputerach kwantowych na rzecz praktycznej obsługi magazynów kluczy (Vaults), zarządzania cyklem życia certyfikatów SSL/TLS oraz zabezpieczania sesji SSH.
3) Dyferencjacja INF.11.2 i INF.12.2 - spójne podstawy powinny stanowić mniejszy procent jednostki, a pozostałe godziny należy przeznaczyć na profilowanie (bezpieczeństwo sieci vs bezpieczeństwo aplikacji).
4) Uzupełnienie OSR - należy wprost wskazać, że realizacja tych jednostek w sposób inny niż teoretyczny wymaga zakupu licencji na specjalistyczne skanery podatności, systemy SIEM oraz dostęp do środowisk chmurowych (IaaS/PaaS), co generuje realne koszty rzędu tysięcy złotych na stanowisko.

4.3. Dotyczy INF.11.3. Programowanie w języku Python

4.3.1. Analiza jednostki INF.11.3

Analiza jednostki INF.11.3. Programowanie w języku Python (120h) w kontekście profilu ucznia technikum (15-17 lat) budzi zastrzeżenia w zakresie przeskalowania niektórych kryteriów. Choć Python jest fundamentem automatyzacji, autorzy podstawy włączyli do niej zagadnienia z zakresu inżynierii oprogramowania i zaawansowanej matematyki, które wykraczają poza "niezbędne minimum" dla technika na poziomie PRK 4.
Można wyróżnić kilka obszarów:

1) Nadmiarowość koncepcyjna i programistyczna - pewne mechanizmy języka są zbyt abstrakcyjne dla poziomu operacyjnego technika cyberbezpieczeństwa:
    - Stosowanie domknięć (closures) oraz generatorów - o ile generatory są przydatne przy dużych logach, o tyle "closures" to koncept z programowania funkcyjnego, który rzadko znajduje bezpośrednie zastosowanie w prostych skryptach administracyjnych.
    - Implementacja programów asynchronicznych (asyncio) - programowanie asynchroniczne jest trudne do debugowania i zrozumienia dla początkujących. Wymaganie go jako standardu do pobierania danych z usług webowych jest przesadą – dla technika wystarczy znajomość programowania sekwencyjnego lub wielowątkowego (threading).
    - Stosowanie iteracji i rekurencji - rekurencja jest konceptem akademickim, który w skryptach systemowych często prowadzi do błędów (przepełnienie stosu) i jest trudny do opanowania przez uczniów bez silnego zaplecza algorytmicznego.
2) Przeładowanie matematyczno-logiczne: prawa de Morgana i konwersja liczb niedziesiątkowych - system binarny i szesnastkowy to podstawa, ale wymaganie sprawnego operowania prawami de Morgana w kodzie Python na poziomie 120-godzinnego kursu "narzędziowego" jest nadmiarowe względem realnych potrzeb utwardzania systemów.
3) Zbyt szeroki zakres automatyzacji i analizy danych - jednostka ta próbuje zrobić z technika specjalistę Data Science, realizowaną najczęściej na uczelniach na studiach II stopnia:
    - Wykorzystanie bibliotek Pandas i NumPy do transformacji dużych zbiorów danych - to są potężne narzędzia wymagające osobnych kursów. Dla technika cyberbezpieczeństwa wystarczające byłoby operowanie na prostych plikach tekstowych i formacie JSON za pomocą standardowych bibliotek.

- Korelacja danych z różnych źródeł w celu wykrywania anomalii - jest to zadanie na poziomie inżynierskim lub dla systemów klasy SIEM, a nie do realizacji w programie nauczania dla ucznia technikum.

4) Ryzyko “AI-dependency”: wykorzystanie AI do generacji testów na podstawie docstringów - podstawa nakłada na ucznia obowiązek korzystania z AI, zanim ten opanuje solidne podstawy manualnego testowania. Może to prowadzić do sytuacji, w której uczeń nie rozumie kodu, który "wygenerował i przetestował" za pomocą AI.

Podsumowując, jednostka INF.11.3 (Programowanie w języku Python) w obecnym brzmieniu jest przeładowana koncepcyjnie. Około 15% kryteriów, takich jak programowanie asynchroniczne (asyncio), stosowanie domknięć (closures) oraz zaawansowana transformacja dużych zbiorów danych (Pandas, NumPy), jest zbyt zaawansowanych dla uczniów technikum w wieku 15-17 lat. Są to zagadnienia typowe dla studiów inżynierskich lub zawodowych deweloperów, a nie dla technika cyberbezpieczeństwa, który ma wykorzystywać Python głównie do prostej automatyzacji i skryptowania zadań systemowych. Proponujemy ograniczenie tych zapisów na rzecz solidnego opanowania bibliotek sieciowych i obsługi plików, co jest realnym “niezbędnym minimum” w tym zawodzie.

4.3.2. Uwagi dotyczące wybranych efektów kształcenia INF.11.3

Efekt kształcenia	**Kryterium weryfikacji**	**Uwagi**
2) implementuje proste programy w języku Python	2) implementuje proste algorytmy sortowania, wyszukiwania	Propozycja wymiany kilku, aby pokazać, które są oczekiwane, a które już nie - przymiotnik "proste" jest subiektywny.
2) implementuje proste programy w języku Python	3) korzysta z funkcji, wyrażeń lambda, domknięć (closures) oraz generatorów	Domknięcia są zbyt trudne do zrozumienia dla części programistów, a co dopiero dla techników. Jest to niepotrzebne na tym etapie kształcenia. Propozycja usunięcia tej część KW.
3) implementuje zaawansowane programy w języku Python	4) implementuje programy asynchroniczne (asyncio) (np. do pobierania danych z wielu usług webowych)	Za trudne jako wymaganie powszechne. To opanują tylko najambitniejsi uczniowie, wymaganie od wszystkich zrozumienia asynchroniczności na, de

		facto, podstawach programowania, jest nierealne. Propozycja usunięcia.
4) tworzy poprawne i bezpieczne programy w języku Python	3) zabezpiecza program przed wyciekami pamięci	To Python, tutaj nie zarządzamy pamięcią jak w C++. Propozycja uszczegółowienia wskazujące na to, aby zwalniać zasoby, usuwać niepotrzebne referencje czy kontrolować przyrost struktur danych.
4) tworzy poprawne i bezpieczne programy w języku Python	4) stosuje zasady bezpiecznego programowania: (...) b) używanie właściwych typów danych do przechowywania sekretów	Wydaje się mało jednoznaczne. Aby uniknąć tego, że ktoś dopisze do tego KW użycie chmurowych usług do przechowania sekretów, proponuje się zapis: “b) wykorzystanie plików konfiguracyjnych i zmiennych środowiskowych do przekazywania sekretów"
4) tworzy poprawne i bezpieczne programy w języku Python	5) wykonuje automatyczne testy oprogramowania	Chodzi o testy jednostkowe czy pełne TDD? Należy doprecyzować.
4) tworzy poprawne i bezpieczne programy w języku Python	6) wykorzystuje narzędzia do weryfikacji programów Python pod kątem obecności błędów bezpieczeństwa	To może być zbyt ambitne na tym etapie.
4) tworzy poprawne i bezpieczne programy w języku Python	7) wykorzystuje narzędzia sztucznej inteligencji do: (...) c) weryfikacji programów pod kątem poprawności i bezpieczeństwa	Proponuje się zapis: “c) pomocniczej weryfikacji…” Uczeń na etapie podstaw programowania naprawdę może nie rozumieć jeszcze w pełni mechanizmów poprawności i

		bezpieczeństwa, więc realnie będzie mieć problem, by ocenić, czy to, co proponuje AI jest w porządku.
5) wykorzystuje język Python do przetwarzania danych oraz automatyzacji działań operacyjnych w cyberbezpieczeństwie	2) wykrywa proste anomalie i incydenty (np. skanowanie portów, eksfiltrację danych) poprzez korelację danych z różnych źródeł	Zbyt trudne dla typowego ucznia. To już jest krok do pracy analityka SOC. Elementy tego można pokazać, ale nie możemy oczekiwać od uczniów, że będą w stanie korelować wiele źródeł danych jednocześnie. Proponuje się usunąć.
5) wykorzystuje język Python do przetwarzania danych oraz automatyzacji działań operacyjnych w cyberbezpieczeństwie	4) automatyzuje proces zbierania informacji oraz weryfikacji podatności przy użyciu bibliotek sieciowych (np. Requests, Socket, Scapy)	Dość ambitne. Może dopisać, że "z użyciem narzędzi sztucznej inteligencji"?
5) wykorzystuje język Python do przetwarzania danych oraz automatyzacji działań operacyjnych w cyberbezpieczeństwie	5) buduje skrypty integrujące różne narzędzia bezpieczeństwa poprzez ich interfejsy API (np. VirusTotal, Shodan, systemy Security Information and Event Management (SIEM))	Za trudne dla przeciętnego ucznia. Jesteśmy na etapie podstaw programowania, klasy 1-2. Komunikacja po API z zewnętrznymi usługami jest złożona. Pojedyncze proste przykłady przejdą, ale wymiana informacji i integracja narzędzi przez API na tym etapie jest nierealna. Proponuje się zapis: “5) buduje proste skrypty odpytujące podstawowe narzędzia bezpieczeństwa (np. VirusTotal, Shodan) poprzez ich interfejsy API"

4.4. Dotyczy INF.11.4. Projektowanie i administracja relacyjnymi bazami danych

4.4.1. Analiza jednostki INF.11.4

1. Uwaga generalna - zmiana przypisania INF.11.4 do kwalifikacji INF.12
Jak wspomniano w pkt. 3.2 niniejszego opracowania, w kontekście analizy przypisanych jednostek efektów kształcenia do danej kwalifikacji:
obecna jednostka efektów kształcenia INF.11.4. Projektowanie i administracja relacyjnymi bazami danych" jest przypisana do kwalifikacji INF.11, Natomiast powinna zostać przypisana do kwalifikacji INF.12.

2. Błędy merytoryczne i brak neutralności technologicznej
Projekt wykazuje niespójność w podejściu do wyboru narzędzi, co stoi w sprzeczności z ogólnymi zasadami tworzenia podstaw programowych:

- Wskazywanie konkretnych silników bazodanowych - w kryteriach weryfikacji wprost wymieniono systemy PostgreSQL (np. funkcja COPY FROM PROGRAM) oraz MongoDB. Takie sformułowanie zapisów zmusza szkoły do zakupu/instalacji konkretnego oprogramowania, zamiast skupiać się na uniwersalnych standardach SQL/NoSQL.
- Mieszanie standardów bezpieczeństwa - kryteria dotyczące wyłączania protokołów SSL 3.0 czy TLS 1.0/1.1 są bardzo szczegółowe, ale ich umieszczenie w sekcji dotyczącej baz danych (zamiast w ogólnych standardach sieciowych) świadczy o braku spójnej wizji architektury bezpieczeństwa w całym projekcie.

3. Uchybienia metodyczne w zakresie wykorzystania Sztucznej Inteligencji (AI)
Jednostka wprowadza wymóg korzystania z narzędzi generatywnej AI do generowania złożonych zapytań SQL oraz ich refaktoryzacji.

- Zagrożenie dla procesu dydaktycznego: Wprowadzenie AI na etapie nauki podstaw języka SQL (efekt 4) jest błędem metodycznym. Uczeń, który ma „weryfikować ręcznie kod wygenerowany przez AI", nie posiadając jeszcze ugruntowanej wiedzy o logice zapytań, będzie jedynie biernym odbiorcą technologii, co uniemożliwi mu wykształcenie samodzielności myślenia algorytmicznego.
- Ryzyko „pójścia na skróty": AI staje się tu protezą, która maskuje brak umiejętności pisania poprawnego kodu SQL, co jest krytyczne w pracy technika cyberbezpieczeństwa.

3. Błędy dydaktyczne: Przeładowanie i chaos poziomów PRK
Analiza gradacji trudności kryteriów wykazuje brak zrozumienia dla możliwości percepcyjnych ucznia technikum w ramach 150-godzinnego modułu:

- Nierealny zakres materiału - w jednej jednostce połączono: podstawy SQL, zaawansowane mechanizmy transakcyjne, administrację serwerami, backup

3-2-1, zaawansowane utwardzanie (hardening), kryptografię danych „at-rest” oraz obsługę baz NoSQL.

- Brak spójności poziomów - obok prostych definicji (np. „wyjaśnia pojęcie tabela”) występują zadania na poziomie eksperckim, takie jak „implementacja modelu ABAC (Attribute-Based Access Control)” czy „konfiguracja kluczy sprzętowych U2F”. Takie zestawienie narusza hierarchię poziomów PRK, czyniąc podstawę niemożliwą do rzetelnej realizacji w założonym czasie.

4. Bariery egzaminacyjne: Dokumentacja zamiast Wykonania (Model D)
Mimo że nazwa jednostki sugeruje „projektowanie i administrację”, przeważająca liczba kryteriów weryfikacji opiera się na czasownikach pasywnych:

- Przykłady teoretyzacji „opisuje strukturę biznesplanu” (w module bazodanowym!), „wyjaśnia pojęcie incydentu”, „wymienia cele działań”.
- Implikacje dla CKE - taka konstrukcja wymusza na egzaminatorach stosowanie modelu dokumentacyjnego. Absolwent może uzyskać certyfikat, potrafiąc opisać procedurę odtwarzania bazy po awarii, ale nie potrafiąc jej fizycznie przeprowadzić w konsoli systemowej pod presją czasu.

5. Ukryte koszty infrastrukturalne (Błąd w OSR)
Realizacja efektu 6 (mechanizmy autoryzacji) oraz efektu 7 (ochrona danych) wymaga specyficznego wyposażenia:

- Wymagany sprzęt - fizyczne klucze sprzętowe (U2F), serwery z procesorami wspierającymi wirtualizację do izolacji poświadczeń oraz licencje na komercyjne magazyny sekretów (Vaults).
- Sprzeczność z OSR - te wymogi generują realne koszty rzędu tysięcy złotych na każdą pracownię, co całkowicie podważa deklarację legislatorów o „braku skutków finansowych”.

4.4.2. Uwagi dotyczące wybranych efektów kształcenia INF.11.4

Efekt kształcenia	Kryterium weryfikacji	Uwagi
7) implementuje mechanizmy ochrony i monitorowania bezpieczeństwa danych w systemach zarządzania bazami danych	1) wdraża szyfrowanie danych w spoczynku, co chroni pliki bazy danych przed odczytem w przypadku ich fizycznej kradzieży z serwera	Czy podstawa programowa powinna wskazywać uzasadnienie dla KW, tak jak w wielu miejscach w tym JEKu? W innych podstawach programowych raczej nie mamy takich zapisów - przy czym z drugiej strony może lepiej opisać, o co chodzi, zamiast jak w INF.04 opisać wszystko tak ogólnie, że podpiąć

		można pod KW wszystko... To ogólna uwaga do tego całego JEKu.
7) implementuje mechanizmy ochrony i monitorowania bezpieczeństwa danych w systemach zarządzania bazami danych	4) opracowuje automatyczne reguły alarmowe, (np. wysyłanie powiadomień na kanał komunikatora technicznego lub e-mail)natychmiast po wykryciu krytycznych błędów bazy	Nadmiarowy przecinek po "alarmowe", brak spacji po zamknięciu nawiasu
7) implementuje mechanizmy ochrony i monitorowania bezpieczeństwa danych w systemach zarządzania bazami danych	5) blokuje nieużywane funkcje sieciowe i porty na poziomie silnika bazy danych, aby zminimalizować powierzchnię potencjalnego ataku: (...) b) blokuje zbędne protokoły sieciowe oraz ogranicza nasłuchiwanie usługi wyłącznie do wymaganych interfejsów, (np. bindowanie do adresu localhost lub konkretnego VLAN-u) zamiast wszystkich dostępnych interfejsów	Nadmiarowy przecinek po "interfejsów"
7) implementuje mechanizmy ochrony i monitorowania bezpieczeństwa danych w systemach zarządzania bazami danych	5) blokuje nieużywane funkcje sieciowe i porty na poziomie silnika bazy danych, aby zminimalizować powierzchnię potencjalnego ataku: (...) c) wyłącza wbudowane komponenty rozszerzające, takie jak moduły do bezpośredniej wysyłki	Brak przecinka przed "jeśli"

	e-mail z bazy czy obsługa skryptów zewnętrznych (np. w językach Python/R) jeśli nie są one niezbędne dla aplikacji	
7) implementuje mechanizmy ochrony i monitorowania bezpieczeństwa danych w systemach zarządzania bazami danych	5) blokuje nieużywane funkcje sieciowe i porty na poziomie silnika bazy danych, aby zminimalizować powierzchnię potencjalnego ataku: (...) g) ewaluuje protokoły i szyfry komunikacyjne, wymuszając stosowanie bezpiecznych wersji TLS oraz wyłączając podatne na ataki, przestarzałe standardy (np. SSL 3.0, TLS 1.0/1.1)	Uczeń ma ewaluować jakość protokołów i szyfrów, czy tylko aktualizować listy protokołów używanych przez aplikację zgodnie z zaleceniami branżowymi? Jeśli to drugie, może lepiej zapisać "g) konfiguruje użycie bezpiecznych wersji TLS[, wyłącza z użycia przestarzałe standardy[(np. ...)]]"
8) identyfikuje i przeciwdziała podatnościom typu SQL Injection	3) stosuje parametryzację zapytań w celu ochrony bazy	Ten JEK mówi o bazach danych, nie o programowaniu - czy nie należy tego EK co najmniej częściowo przenieść np. do "Programowania w języku Python"?
8) identyfikuje i przeciwdziała podatnościom typu SQL Injection	4) wdraża walidację i sanityzację danych wejściowych od użytkownika	Ten JEK mówi o bazach danych, nie o programowaniu - czy nie należy tego EK co najmniej częściowo przenieść np. do "Programowania w języku Python"?
8) identyfikuje i przeciwdziała podatnościom typu SQL Injection	6) przeprowadza testy bezpieczeństwa pod kątem obecności luk typu injection	Ten JEK mówi o bazach danych, nie o programowaniu - czy nie należy tego EK co najmniej częściowo przenieść np. do "Programowania w języku

		Python"?

Rekomendacje:

1) Usunięcie nazw własnych oprogramowania (PostgreSQL, MongoDB) z kryteriów weryfikacji na rzecz wymagań funkcjonalnych.
2) Ograniczenie roli AI w procesie nauki podstaw SQL – narzędzia te powinny być wprowadzone dopiero w klasach programowo najwyższych.
3) Wydzielenie administracji bazami danych jako osobnej, zaawansowanej jednostki, aby uniknąć przeładowania poznawczego.
4) Urealnienie kryteriów weryfikacji poprzez zastąpienie „wyjaśnia/opisuje” czasownikami „konfiguruje/implementuje”, co pozwoli na przeprowadzenie realnego egzaminu wykonawczego.

4.5. Dotyczy INF.11.5. Administrowanie sieciami TCP/IP

4.5.1. Analiza jednostki INF.11.5

1. Rażąca sprzeczność między wymogami sprzętowymi a Oceną Skutków Regulacji (OSR)

Jednostka INF.11.5 nakłada na szkoły najbardziej kosztowne wymagania w całym programie technika cyberbezpieczeństwa.

- Wymagany sprzęt - szkoła musi zapewnić „zestaw zarządzalnych fizycznych przełączników (warstwa 2 i 3) oraz routerów (jeden zestaw na dwóch uczniów)” oraz „sprzętową zaporę sieciową (firewall)”.
- Analiza kosztów - przy 30-osobowej klasie oznacza to konieczność zakupu 15 profesjonalnych zestawów sieciowych. Koszt jednego zestawu (2 przełączniki zarządzalne + router klasy biznesowej) to wydatek rzędu 5 000 – 10 000 zł. Łączny koszt doposażenia tylko jednej pracowni wynosi od 75 000 do 150 000 zł, nie licząc sprzętowej zapory ogniowej.
- Wniosek - utrzymanie w OSR zapisu o „zerowych kosztach dla budżetu państwa” i „trudnościach w oszacowaniu kosztów dla samorządów” jest w przypadku tej jednostki dowodem na całkowite oderwanie projektu od realiów rynkowych i cen profesjonalnego sprzętu sieciowego.

2. Metodyczna teoretyzacja kompetencji (Model D zamiast Modelu W)

Mimo że nazwa jednostki to „Administrowanie”, co sugeruje działanie, znaczna część kryteriów weryfikacji jest sformułowana w sposób pasywny:

- Pasywne kryteria - uczeń ma jedynie: „wyjaśniać pojęcie”, „opisywać różnice”, „wymieniać uczestników”, „rozpoznawać adresy” .
- Problem egzaminacyjny - taka konstrukcja kryteriów uniemożliwia rzetelną walidację umiejętności na egzaminie zawodowym. Zamiast fizycznej konfiguracji urządzeń na stanowisku egzaminacyjnym (Model W), CKE będzie

zmuszona do przygotowania zadań polegających na opisywaniu konfiguracji lub interpretacji zrzutów ekranu (Model D) .
- Skutek - absolwent może uzyskać dyplom państwowy bez umiejętności fizycznego podłączenia i zabezpieczenia sieci, co czyni go bezużytecznym dla pracodawców z branży IT.

3. Błędy dydaktyczne: Przeładowanie programowe i brak gradacji (PRK)
Jednostka przewiduje 150 godzin na opanowanie materiału, który w standardach rynkowych (np. ścieżka certyfikacji Cisco CCNA) zajmuje znacznie więcej czasu i wymaga setek godzin praktyki.

- Zbyt szeroki zakres - w jednym module połączono: podstawy teorii sygnałów, adresację IPv4 i IPv6, konfigurację przełączników L2/L3, routing statyczny i dynamiczny, segmentację VLAN, listy kontroli dostępu ACL, translację NAT oraz – co najbardziej nierealne w tym czasie – konfigurację tuneli VPN (OpenVPN, WireGuard, IPsec) i sprzętowych firewalli.
- Brak spójności poziomów - obok elementarnej wiedzy (np. „wyjaśnia cel wymiany danych") występują zadania z poziomu inżynierskiego, takie jak „konfiguracja bezpiecznego dostępu do sieci bezprzewodowych standardu WPA3" czy „zarządzanie trasowaniem dynamicznym". Taki chaos utrudnia rzetelną realizację programu.

4. Anachronizm merytoryczny i braki w nowoczesnych technologiach
Mimo aspiracji do zawodu „Cyberbezpieczeństwo", jednostka pomija kluczowe, nowoczesne trendy w administracji sieciami:

- Brak ZTNA i SDN - podstawa nie wspomina o architekturze Zero Trust Network Access (ZTNA) ani o sieciach definiowanych programowo (Software-Defined Networking – SDN), które są obecnie standardem w ochronie infrastruktur krytycznych.
- Skupienie na technologiach schodzących - nadmierny nacisk na konfigurację „tradycyjnych" usług, przy jednoczesnym unikaniu wskazywania konkretnych narzędzi do monitorowania bezpieczeństwa warstwy 2 (np. ochrona przed atakami typu ARP Spoofing czy DHCP Starvation), czyni kształcenie częściowo anachronicznym.

4.5.2. Uwagi dotyczące wybranych efektów kształcenia INF.11.5

Efekt kształcenia	Kryterium weryfikacji	Uwagi
1) wyjaśnia podstawowe pojęcia i zasady funkcjonowania sieci komputerowych	6) opisuje topologie sieci komputerowych (np. gwiazdy, magistrali, pierścienia) oraz wskazuje ich zalety i ograniczenia	Może czas już skończyć z topologią magistrali, a i kontrowersyjna topologia pierścieniowa to też raczej powód do sporów niż realna wiedzowa

		potrzeba
2) charakteryzuje działanie sieci komputerowych w oparciu o modele OSI/TCP-IP oraz protokoły transmisji danych	1) opisuje warstwy modeli ISO/OSI i TCP/IP oraz przypisuje im protokoły	Od dawna używamy pojęcia "model OSI" Jakie protokoły - wszystkich warstw, czy skupiamy się tylko na najwyższych warstwach? Należy doprecyzować
2) charakteryzuje działanie sieci komputerowych w oparciu o modele OSI/TCP-IP oraz protokoły transmisji danych	3) opisuje budowę adresu IPv4 (adres IP, maska, adres sieci, adres rozgłoszeniowy)	NIepotrzebnie powtórzony “adres IP” w nawiasie. Do usunięcia.
2) charakteryzuje działanie sieci komputerowych w oparciu o modele OSI/TCP-IP oraz protokoły transmisji danych	7) wskazuje zależność pomiędzy protokołami warstwy sieciowej i transportowej, a protokołami warstwy aplikacji (na przykładzie HTTP)	Obecnie protokół HTTP jest w mniejszości.
2) charakteryzuje działanie sieci komputerowych w oparciu o modele OSI/TCP-IP oraz protokoły transmisji danych	8) wyjaśnia rolę protokołu DNS w procesie translacji nazw adresów	Powinno być nazw domenowych na adres IPv4 i IPv6
2) charakteryzuje działanie sieci komputerowych w oparciu o modele OSI/TCP-IP oraz protokoły transmisji danych	10) opisuje pojęcie sniffingu sieciowego oraz przeznaczenie narzędzia Wireshark jako analizatora ruchu sieciowego, z uwzględnieniem zastosowań diagnostycznych i bezpieczeństwa	To kryterium nie pasuje do jednostki efektu kształcenia.
3) stosuje adresację IPv4 i IPv6	1) rozpoznaje adresy IPv4 i IPv6 oraz określa ich budowę, w tym zapis adresów z wykorzystaniem notacji Classless Inter-Domain Routing (CIDR)	Częściowo powtórzony INF.11.5.E2.K3

3) stosuje adresację IPv4 i IPv6	3) identyfikuje adres sieci, adres rozgłoszeniowy i adres hosta na podstawie adresu IP i notacji CIDR	Należy doprecyzować, że mowa o IPv4.
3) stosuje adresację IPv4 i IPv6	5) wyjaśnia znaczenie adresacji IP dla bezpieczeństwa i identyfikacji źródeł ruchu	O jakim adresie mowa IPv4 i IPv6.
4) konfiguruje podstawowe parametry sieciowe urządzeń końcowych	6) wyjaśnia wpływ błędnej konfiguracji hosta na bezpieczeństwo sieci	Co autor miał na myśli ? tu można napisać 4 doktoraty i dwie habilitacje. Bardzo ogólne i enigmatyczne stwierdzenie
6) przeprowadza podstawową konfigurację podsieci	1) wyjaśnia różnice między przełączaniem a routowaniem	1. Zapis "przeprowadza podstawową konfigurację podsieci" sugeruje, że uczeń konfiguruję podsieć w jakieś sieci, nie powinno być "przeprowadza podstawową konfigurację sieci lokalnej" 2. Należy rozdzielić na konfigurację przełączników oraz routerów.
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje podstawowe parametry przełączników i routerów, w tym:	1. Kolejność wprost wyjęta z kolejnych kroków konfiguracji urządzeń Cisco. 2. Co oznacza "podstawowe" parametry? Brak podanego zakresu.
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje podstawowe parametry przełączników i routerów, w tym: (...) a) sprawdza aktualną konfigurację	A czy nie powinien również analizować konfigurację urządzenia?

	urządzenia sieciowego	
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje podstawowe parametry przełączników i routerów, w tym: (...) d) przypisuje adres IP do interfejsu routera	Trzeba doprecyzować, że przypisuje adresy IPv4 i IPv6.
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje podstawowe parametry przełączników i routerów, w tym: (...) g) sprawdza zawartość tablicy routingu oraz interpretuje podstawowe wpisy (sieć lokalna, trasa domyślna)	Aby interpertować wpisy w tablicy routingu należy wprowadzić pojęcie routingu dynamicznego oraz parametrów z nimi związanych.
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje bezpieczny dostęp do sieci bezprzewodowych, w tym:	Błąd kolejności numeracji kryteriów weryfikacji - powinno być 4)
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje bezpieczny dostęp do sieci bezprzewodowych, w tym: (...) b) standard zabezpieczenia (WPA2/WPA3)	Należy unikać konkretnych standardów, a jak za rok powstanie nowy standard - musimy mieć możliwość wprowadzania nowych technologii.
6) przeprowadza podstawową konfigurację podsieci	3) konfiguruje bezpieczny dostęp do sieci bezprzewodowych, w tym: (...) d) wyłączenie funkcji WPS	Obecnie routery w większości nie posiadają tej funkcji. Brak wyjaśnienia dlaczego należy wyłączyć
7) stosuje mechanizmy segmentacji i kontroli ruchu w sieciach TCP/IP	1) wyjaśnia znaczenie segmentacji sieci dla bezpieczeństwa, (...) a) topologia gwiazdy	Niepotrzebne powielanie treści, to już było w INF.11.5.E1.K6

7) stosuje mechanizmy segmentacji i kontroli ruchu w sieciach TCP/IP	1) wyjaśnia znaczenie segmentacji sieci dla bezpieczeństwa, (...) c) router brzegowy, rdzeniowy, agregacyjny	“wyjaśnia pojęcia…”, ale routing w osobnym JEK, masło maślane, warstwa 2, ACL czyli być może i 4, zapora sieciowa i VPN a routing wyrzucony do JEK 8.
7) stosuje mechanizmy segmentacji i kontroli ruchu w sieciach TCP/IP	4) konfiguruje połączenia typu trunk	Nie wszyscy producenci używają pojęcia trunk chyba wiadomo skąd to - ruch tagowany 802.1q
7) stosuje mechanizmy segmentacji i kontroli ruchu w sieciach TCP/IP	6) konfiguruje podstawowe reguły sprzętowej zapory sieciowej	Jakie to są podstawowe reguły? A kiedy już mowa o rozszerzonych?
7) stosuje mechanizmy segmentacji i kontroli ruchu w sieciach TCP/IP	7) konfiguruje podstawowe połączenia VPN, (np. OpenVPN, WireGuard, IPsec)	Konfiguruje połączenia VPN, tylko nie wie co to jest, jak działa, do czego służy i czym się poszczególne typy różnią
8) konfiguruje mechanizmy routingu i translacji	2) rozpoznaje protokoły routingu dynamicznego i opisuje ich zastosowanie	Na jakiej podstawie rozpoznaje protokoły routingu po nazwie, działaniu?
8) konfiguruje mechanizmy routingu i translacji	3) konfiguruje translację adresów sieciowych (NAT) oraz translację numerów portów (port forwarding)	Brak wprowadzenia do tematu czym jest NAT i translacja.
8) konfiguruje mechanizmy routingu i translacji	4) wyjaśnia wpływ NAT na bezpieczeństw o i analizę ruchu sieciowego	Brak podstaw działania NAT.

Rekomendacje

1) Urealnienie czasowników operacyjnych - zastąpienie „wyjaśnia/opisuje” czasownikami „konfiguruje”, „zabezpiecza”, „monitoruje”, „testuje”. Egzamin musi odbywać się na fizycznym sprzęcie, a nie w formie testu pisemnego.
2) Korekta OSR - należy wprost przyznać, że ta jednostka wymaga dedykowanej dotacji celowej dla szkół na zakup sprzętu sieciowego L3 oraz sprzętowych rozwiązań UTM/Firewall.
3) Wydzielenie bezpieczeństwa sieciowego - sugeruje się wydzielenie najbardziej zaawansowanych tematów (VPN, sprzętowe firewalle, IDS/IPS)

do osobnej jednostki w wyższej klasie, aby zapewnić uczniom czas na rzetelne opanowanie podstaw przełączania i rutowania.

4) Zestawienie z certyfikatami rynkowymi - warto dodać zapis o dążeniu do zgodności efektów kształcenia z międzynarodowymi standardami (np. CompTIA Network+ lub CCNA), co podniosłoby rangę dyplomu ponad poziom teoretycznego opisu.

4.6. Dotyczy INF.11.6. Użytkowanie, konfiguracja i utwardzanie systemu Windows i Linux

4.6.1. Analiza jednostki INF.11.6

1. Zauważone trudności

Efekty kształcenia i kryteria weryfikacji w INF.11.6 są rzetelne technicznie, ale zbyt trudne dla uczniów technikum. Wiele zapisów (50-60%) jest realnych, ale pozostała część wymaga albo naprawdę ambitnej młodzieży i nauczycieli, albo wydaje się być przeniesiona z poziomu administratora systemów, szkoleń specjalistycznych czy laboratoriów certyfikacyjnych.

Najbardziej problematyczne obszary to:

- zaawansowana analiza architektury systemów operacyjnych,
- rozbudowane zarządzanie tożsamością i ochroną poświadczeń,
- część KW dot. hardeningu systemów operacyjnych,
- pełna weryfikacja zabezpieczeń kontenerów,
- migracje pomiędzy platformami wirtualizacyjnymi,
- zaawansowane usługi chmurowe (IAM, kontenery, sekrety, prywatne rejestry, skanowanie obrazów).

W obecnym brzmieniu zapis łączy zatem podstawy administracji, hardening, elementy forensics i detekcji zagrożeń, IAM, kontenery i cloud security - to za dużo jak na jeden JEK w wieku uczniów szkół średnich. Realnie jest tu zbyt dużo zbyt trudnego materiału.

W tym JEK należy uporządkować zapisy pod kątem kolejnych efektów kształcenia: budowa rozruch systemu, dyski/partycje/instalację/konfigurację zabezpieczeń.

2. Rażące przeładowanie programowe i nierealny zakres merytoryczny

Jednostka INF.11.6 zakłada opanowanie w ciągu 180 godzin pełnej administracji oraz zaawansowanego utwardzania (hardeningu) dwóch skrajnie różnych rodzin systemów operacyjnych: Windows i Linux.

- Dualizm systemowy - uczeń musi w jednym module opanować m.in. zasady grup GPO, rejestr i AppLocker (Windows) oraz SELinux/AppArmor, moduły PAM i systemy audytu auditd (Linux). W praktyce rynkowej są to dwie odrębne specjalizacje.
- Zakres "od biosu po chmurę" - podstawa obejmuje procesy rozruchu, systemy plików, uprawnienia, usługi, harmonogramy zadań, rejestry zdarzeń, złośliwe

oprogramowanie, wirtualizację, konteneryzację (Docker) oraz wdrażanie bezpiecznych środowisk w chmurze (IaaS, PaaS, SaaS) .
- Wniosek - tak szeroki zakres materiału wymusza na nauczycielach realizację pobieżną i encyklopedyczną, co stoi w sprzeczności z celem kształcenia specjalisty ds. cyberbezpieczeństwa, który musi rozumieć niskopoziomowe detale systemowe.

3. Metodyczne ryzyko związane z przedwczesnym wprowadzaniem AI
Efekt kształcenia nr 6 wprost nakazuje wykorzystywanie narzędzi Sztucznej Inteligencji do generowania skryptów PowerShell i Bash oraz interpretowania błędów systemowych.
- Zagrożenie dla fundamentów wiedzy - dopuszczenie generowania kodu przez AI na etapie nauki podstaw administracji (zanim uczeń zrozumie składnię i logikę systemową) doprowadzi do wykształcenia postawy bezkrytycznego kopiowania rozwiązań.
- Bezpieczeństwo kodu - kryteria wspominają o weryfikacji kodu z AI, ale uczeń bez ugruntowanej wiedzy technicznej nie jest w stanie rzetelnie ocenić, czy wygenerowany skrypt nie zawiera luk bezpieczeństwa lub błędów logicznych.

4. Sprzeczność wymagań infrastrukturalnych z Oceną Skutków Regulacji (OSR)
Jednostka ta generuje potężne zapotrzebowanie na zasoby sprzętowe i licencje, co zostało całkowicie zignorowane w finansowej części projektu.
- Wymagania sprzętowe - do realizacji laboratoriów z wirtualizacji i konteneryzacji wymagane są stacje robocze o zasobach min. 16 GB RAM umożliwiające równoległe uruchamianie wielu systemów i narzędzi testowych.
- Koszty chmurowe - podstawa wymaga zapewnienia dostępu do platform chmurowych (publicznych lub hybrydowych) oraz konsol administracyjnych do zarządzania bezpieczeństwem w chmurze.
- Błąd w OSR - twierdzenie, że wejście w życie rozporządzenia nie spowoduje skutków finansowych dla budżetu państwa, jest niemożliwe do obrony, biorąc pod uwagę, że większość polskich szkół nie dysponuje sprzętem o takich parametrach ani budżetami na subskrypcje chmurowe.

5. Pozorność weryfikacji w obszarze "utwardzania" (Hardeningu)
Mimo że "utwardzanie" figuruje w nazwie jednostki, wiele kryteriów weryfikacji sprowadza się do wiedzy deklaratywnej:
- Teoretyczne podejście - uczeń ma "wyjaśniać różnicę", "identyfikować modele", "opisywać rolę" lub "charakteryzować procedury".
- Problem egzaminacyjny - brak ostrego rozdzielenia czynności konfiguracyjnych od opisowych sprawi, że egzamin zawodowy w modelu "D" (dokumentacja) będzie polegał na zaznaczeniu na papierze, co należy zrobić, zamiast na fizycznym "utwardzeniu" systemu w środowisku laboratoryjnym pod kątem konkretnych standardów (np. CIS Benchmarks).

6. Niekompletność w zakresie nowoczesnych zagrożeń
W module dotyczącym ochrony przed szkodliwym oprogramowaniem (efekt 12) skupiono się na systemach antywirusowych i symulacji ataku "reverse shell".Braki merytoryczne: Pominięto kluczowe dla współczesnego cyberbezpieczeństwa systemy klasy EDR (Endpoint Detection and Response) oraz mechanizmy analizy behawioralnej, które są standardem w ochronie systemów Windows i Linux w przemyśle.

Rekomendacje:

1) Rozdzielenie systemów operacyjnych - sugeruje się podział tej jednostki na dwa odrębne moduły (Administracja Windows oraz Administracja Linux), co pozwoli na rzetelne zgłębienie specyfiki każdego z nich.
2) Urealnienie roli AI - narzędzia AI powinny być wprowadzone jako element wspomagający dopiero po zaliczeniu podstaw skryptowania manualnego w klasach wyższych.
3) Korekta OSR i dotacje celowe - konieczne jest przyznanie samorządom funduszy na modernizację pracowni komputerowych (stacje 16-32 GB RAM) oraz na opłacenie dostępu do profesjonalnych środowisk chmurowych (np. Azure, AWS, Google Cloud).
4) Operacjonalizacja kryteriów - zmiana czasowników z "opisuje/wyjaśnia" na "wdraża/utwardza/sprawdza" w kontekście konkretnych standardów bezpieczeństwa, aby wymusić wykonawczą formę egzaminu zawodowego.

4.6.2. Uwagi dotyczące wybranych efektów kształcenia INF.11.6

Dotyczy:
2) analizuje architekturę systemów operacyjnych pod kątem mechanizmów ochronnych i wektorów ataków

Propozycja zastąpienia obecnego efektu kształcenie efektem, w którym uczeń ma rozumieć, że takie mechanizmy są, a nie musieć znać je ekstremalnie dogłębne - co jest wymagane przy analizie architektury systemów operacyjnych.

Rekomendacja:
EK: 2) opisuje wybrane rozwiązania architektury systemów operacyjnych dotyczące mechanizmów ochronnych i wektorów ataków
KW:
1) opisuje rolę trybu jądra i trybu użytkownika [choć to już jest w 1) EK, więc można tu pominąć]
2) opisuje mechanizm przełączania zadań pomiędzy trybem jądra a trybem użytkownika
3) opisuje wybrane mechanizmy ochrony pamięci (np. ASLR, DEP) i ich rolę

4) opisuje mechanizmy systemów plików NTFS i ext4 dotyczące bezpieczeństwa i poufności danych
5) opisuje rolę bezpieczeństwa łańcucha rozruchu (Secure Boot, Trusted Boot) w kontekście ochrony przed rootkitami
6) opisuje mechanizmy ról i uprawnień do zasobów w środowiskach wielodostępowych
7) wymienia przykładowe historyczne wektory ataków na systemy operacyjne wykorzystujące specyficzne podatności ich architektury (przepełnienie bufora, wyścigi, ataki na mechanizm uprawnień, Spectre / Meltdown)

Efekt kształcenia	**Kryterium weryfikacji**	**Uwagi**
2) analizuje architekturę systemów operacyjnych pod kątem mechanizmów ochronnych i wektorów ataków	6) weryfikuje integralność jądra systemu oraz autentyczność sterowników	Mało realne do zrealizowania przez technika. Proponuje się usunąć.
2) analizuje architekturę systemów operacyjnych pod kątem mechanizmów ochronnych i wektorów ataków	7) identyfikuje wektory ataków wykorzystujące specyficzne podatności architektury systemów operacyjnych	Przykłady da się omówić, o rzetelnej analizie nie ma mowy na tym poziomie kształcenia.
3) wdraża systemy operacyjne w standardzie utwardzonej instalacji	2) przeprowadza partycjonowanie dysków z uwzględnieniem izolacji katalogów systemowych i stosowania flag montowania ograniczających wykonywanie kodu	Stosowanie flag raczej nie będzie osiągalne dla typowego ucznia.
3) wdraża systemy operacyjne w standardzie utwardzonej instalacji	7) aktualizuje system bezpośrednio po instalacji w celu wyeliminowania luk bezpieczeństwa	I to jest poziom technika - rzadko się to zdarza w tym dokumencie, więc warto zaznaczyć, że jest to dobrze rozpisany EK z KW
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	2) projektuje strukturę grup bezpieczeństwa w celu implementacji modelu RBAC (Role-Based Access Control) na poziomie lokalnym (...) e) analizuje hierarchię	To jest już rola inżyniera, nie technika. Proponuje się uprościć do rozumienia mechanizmu dziedziczenia uprawnień i ewentualnego konfigurowania prostych przykładów, a nie dużej hierarchii grup.

	grup pod kątem optymalizacji dziedziczenia uprawnień w systemie	
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	2) projektuje strukturę grup bezpieczeństwa w celu implementacji modelu RBAC (Role-Based Access Control) na poziomie lokalnym (...) h) opracowuje dokumentację projektową opisującą mapowanie ról na konkretne grupy systemowe	"opracowuje" -> "tworzy"? Na etapie technika raczej oczekujemy wypełnienia prostych dokumentów niż wytworzenia pełnych wzorców dokumentów
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	5) konfiguruje mechanizmy kontroli konta użytkownika w systemach Linux i Windows	Jak to interpretować dla Linuxa? W Windowsie mieliśmy kiedyś suwak do ustawienia "agresywności", teraz już chyba go nie ma.
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	6) konfiguruje bezpieczeństwo profili użytkowników, w tym mechanizmami ochrony poświadczeń w pamięci (np. Windows Defender Credential Guard, Linux PAM i systemy izolacji) (...) d) konfiguruje i zarządza zasadami haseł oraz mechanizmami uwierzytelniania wieloskładnikowego w systemie Linux (moduły PAM)	MFA przez PAM nie na poziomie technika, to jest rola inżynierów.
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	6) konfiguruje bezpieczeństwo profili użytkowników, w tym mechanizmami ochrony poświadczeń w pamięci (np. Windows Defender Credential Guard, Linux	Nie na poziomie technika, to jest rola inżynierów.

	PAM i systemy izolacji) (...) e) aktywuje i weryfikuje działanie funkcji Windows Defender Credential Guard, wykorzystującej wirtualizację do izolacji poświadczeń Local Security Authority (LSA)	
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	6) konfiguruje bezpieczeństwo profili użytkowników, w tym mechanizmami ochrony poświadczeń w pamięci (np. Windows Defender Credential Guard, Linux PAM i systemy izolacji) (...) f) wdraża systemy obowiązkowej kontroli dostępu (MAC), takie jak AppArmor lub SELinux, w celu izolacji procesów użytkownika	Nie na poziomie technika, to jest rola inżynierów.
4) wdraża polityki tożsamości i kontroli dostępu zgodnie z modelem zerowego zaufania w systemach Linux i Windows	6) konfiguruje bezpieczeństwo profili użytkowników, w tym mechanizmami ochrony poświadczeń w pamięci (np. Windows Defender Credential Guard, Linux PAM i systemy izolacji) (...) g) programuje przechowywanie danych i preferencji użytkownika w sposób bezpieczny, wykorzystując systemowe magazyny kluczy	Pomieszanie administracji systemem z programowaniem aplikacyjnym. Usunąć / przenieść do podstaw cyberbezpieczeństwa jako opis możliwości.
6) wykorzystuje narzędzia sztucznej inteligencji do zarządzania systemem	1) generuje skrypty PowerShell i Bash przy użyciu narzędzi sztucznej inteligencji	A jak sprawdzi poprawność skryptu - brak podstaw do tego zadania. Nie można ucznia uczyć korzystania z AI bez krytycznego myślenia, uczniowie często szukają rozwiązań

		w AI a następnie metodą kopiuj-wklej konfigurują usługi. Taki absolwent nie ma odpowiedniej wiedzy. Uczmy go myśleć.
6) wykorzystuje narzędzia sztucznej inteligencji do zarządzania systemem	3) interpretuje za pomocą sztucznej inteligencji skomplikowane komunikaty błędów systemowych	Czemu tylko "skomplikowane"? Proponuję usunąć ten przymiotnik, ponieważ "skomplikowanie" jest subiektywne. Może lepiej “złożone” - tylko na etapie KW brakuje definicji na czym ta złożoność miałaby polegać.
6) wykorzystuje narzędzia sztucznej inteligencji do zarządzania systemem	4) weryfikuje poprawność i bezpieczeństwo kodu skryptu wygenerowanego przez sztuczną inteligencję	Jak ma to robić, skoro nie uczymy języka PowerShell? Dopiszmy, że "wykorzystując narzędzia sztucznej inteligencji". Trzeba też zwrócić uwagę, że w tym EK mówimy już o wykorzystaniu AI - czy w każdym KW musimy o tym przypominać?
7) administruje usługami i harmonogramem zadań	3) planuje cykliczne zadania przy użyciu Harmonogramu Zadań (Windows) i cron (Linux)	Nie ma potrzeby zapisu Harmonogramu Zadań wielką literą. Powinno być "harmonogramu zadań".
7) administruje usługami i harmonogramem zadań	4) analizuje zależności między usługami systemowymi oraz identyfikuje powiązania i błędy kaskadowe	Czy powinien to robić technik? Proponuję zastąpić tylko ustalaniem nadrzędności-podrzędności usług.
7) administruje usługami i harmonogramem zadań	5) monitoruje logi generowane przez poszczególne usługi	Jak rozumieć ten KW? Na egzaminie zawodowym wpiszemy sprawdzenie przebiegu "zdający wpatrywał się w logi usługi X przez co najmniej 15 sekund"?
8) utwardza konfigurację systemu Windows	4) konfiguruje zabezpieczenia	Zbyt nieprecyzyjne. Chodzi o TPM, Secure

	sprzętowe	Boot czy jeszcze szerzej? Doprecyzować.
8) utwardza konfigurację systemu Windows	5) wdraża mechanizmy ochrony przed exploitami i atakami na pamięć	Systemy operacyjne mają już domyślnie uruchomione większość takich mechanizmów. Jeśli chcieć jeszcze bardziej utwardzać, to trzeba być świadomym, dlaczego, po co i jak. Zbyt ambitne dla technika. Proponuję usunąć.
8) utwardza konfigurację systemu Windows	8) konfiguruje ustawienia systemu operacyjnego w zakresie prywatności:	Zbyt szczegółowe wymienienie elementów dla Windowsa (od a do q!!!). W ogóle tutaj to wymienienie raczej jest niepotrzebne, ani dla Windowsa, ani dla Linuxa. Proponuje się usunąć.
9) utwardza konfigurację systemu Linux	2) konfiguruje narzędzie sudo	W jakim zakresie? Brak doprecyzowania.
9) utwardza konfigurację systemu Linux	3) zabezpiecza bootloader hasłem i blokuje nieautoryzowany dostęp fizyczny	W jakim sensie "blokuje nieautoryzowany dostęp fizyczny"? Mamy nauczyć uczniów zamykania drzwi serwerowni?
9) utwardza konfigurację systemu Linux	6) wdraża bezpieczną konfigurację stosu TCP/IP	W jakim sensie "bezpieczną"? Trzeba uszczegółowić albo usunąć ten przymiotnik.
9) utwardza konfigurację systemu Linux	7) weryfikuje uprawnienia plików systemowych pod kątem bitów SUID i SGID	To nie jest zadanie technika, proponuję usunąć.
10) zarządza zaporą sieciową i bezpieczeństwem sieciowym hosta	1) konfiguruje reguły przychodzące i wychodzące w Zaporze Windows	"zaporze"
10) zarządza zaporą sieciową i bezpieczeństwem sieciowym hosta	7) wdraża ochronę przed atakami typu Denial of Service na poziomie hosta	Proste limity i filtry może zrobią, ale pełny komplet - to już rola inżyniera. Proponuję zapis "wdraża podstawową ochronę".

11) obsługuje systemy logowania i audytu zdarzeń	1) wykorzystuje narzędzie „Podgląd Zdarzeń” do analizy błędów i ostrzeżeń systemowych	"Podgląd zdarzeń"
13) sprawdza i zabezpiecza niskopoziomową konfigurację systemu	1) modyfikuje ustawienia systemowe w celu wyłączenia podatnych protokołów	Ten EK wydaje się być zbyt ambitny dla uczniów w tej grupie wiekowej, szczególnie w zakresie Linuxa.
13) sprawdza i zabezpiecza niskopoziomową konfigurację systemu	5) monitoruje zmiany w rejestrze pod kątem działań złośliwego oprogramowania	Zbyt ambitny na poziom technika.
14) wykorzystuje wirtualizację i konteneryzację w pracy z systemami	6) zarządza (buduje, wdraża, wersjonuje, monitoruje oraz usuwa) obrazy i steruje pełnym cyklem życia kontenerów	"zarządza [...] obrazami". Pełny cykl życia kontenerów - to już jest chyba zbyt ambitne (chyba że rozumiemy przez to "utwórz kontener, pokorzystaj, usuń")
14) wykorzystuje wirtualizację i konteneryzację w pracy z systemami	7) optymalizuje wykorzystanie zasobów sprzętowych przez systemy wirtualne	Nierealne - to przychodzi z wieloletnim doświadczeniem (chyba że znowu rozumiemy przez to "nie dawaj maszynie za dużo RAMu"). Proponuję usunąć.
14) wykorzystuje wirtualizację i konteneryzację w pracy z systemami	8)	Zbyt nieprecyzyjne. Możemy wymyśleć ogrom reguł, ale to nie technik powinien je dobierać, ani nie powinien musieć wiedzieć o tym, jak wszystkie z nich konfigurować.
14) wykorzystuje wirtualizację i konteneryzację w pracy z systemami	9) migruje systemy między różnymi platformami wirtualizacyjnymi:	To nie jest rola technika, proponuję usunąć - najlepiej całe 9) KW
15) wdraża w sposób bezpieczny środowiska wirtualne oparte na	7) przypisuje instancjom wirtualnym oraz usługom kontenerowym role	To może być nie do zrealizowania z uczniami w tej grupie wiekowej. To

systemie operacyjnym Linux oraz kontenerowe w środowisku chmurowym	tożsamości, unikając stosowania statycznych poświadczeń wewnątrz systemów	jest ważny element IAM, ale zrozumienie tego wymaga doświadczenia i analogii do innych mechanizmów, obycia z nimi.
15) wdraża w sposób bezpieczny środowiska wirtualne oparte na systemie operacyjnym Linux oraz kontenerowe w środowisku chmurowym	8) wdraża środowiska kontenerowe w chmurze (np. usługi typu Container Instances lub zarządzane klastry Kubernetes), dbając o izolację sieciową zasobów	Nierealne na etapie technikum. Ambitne, ale przeszacowane. Proponuję usunąć.
15) wdraża w sposób bezpieczny środowiska wirtualne oparte na systemie operacyjnym Linux oraz kontenerowe w środowisku chmurowym	9) korzysta z prywatnych rejestrów obrazów kontenerowych w chmurze i konfiguruje automatyczne skanowanie obrazów pod kątem podatności	To też jest ambitne. Mieszamy tu trochę narzędzia administracji systemami i chmurami z dodatkowym oprogramowaniem do szukania podatności. Już samo "korzystanie" z prywatnych rejestrów oznacza, że na zajęciach trzeba je będzie utworzyć. Raczej do usunięcia.
15) wdraża w sposób bezpieczny środowiska wirtualne oparte na systemie operacyjnym Linux oraz kontenerowe w środowisku chmurowym	10) konfiguruje chmurowe mechanizmy zarządzania sekretami	Skonfigurowanie OK, ale jak wytłumaczyć uczniom te mechanizmy bez zastosowania ich w praktyce przy tworzeniu aplikacji? Aby "kliknęło", po co to jest.

4.7. Dotyczy INF.11.7. Testowanie bezpieczeństwa sieci TCP/IP

4.7.1. Analiza jednostki INF.11.7

1. Dydaktyczna powierzchowność: Przeładowanie materiału w 90 godzinach
Jednostka przewiduje zaledwie 90 godzin na opanowanie niezwykle szerokiego zakresu kompetencji, które w realiach rynkowych stanowią kilka odrębnych specjalizacji zawodowych.

- Nierealny zakres - w ramach 90 godzin uczeń ma opanować: skanowanie podatności, analizę logów systemowych (syslog), konfigurację systemów IDS/IPS i SIEM, reagowanie na incydenty, analizę ryzyka oraz

zaawansowaną dokumentację analityczną opartą na bazach CWE, CVE i CAPEC.
- Brak czasu na praktykę - przy tak ogromnej ilości wiedzy teoretycznej, realne przeprowadzenie testów penetracyjnych (np. wykrywanie spoofingu IP/MAC czy analiza ruchu w Wireshark) zostanie sprowadzone do demonstracji przez nauczyciela, zamiast samodzielnego wypracowania nawyków przez ucznia.

2. Błędy merytoryczne i brak hierarchii poziomów (PRK)
Analiza jednostki wykazuje chaos w gradacji trudności zadań, co narusza zasady Polskiej Ramy Kwalifikacji.
- Mieszanie kompetencji - w jednym bloku efektów (Efekt 5) uczeń ma jednocześnie „dokumentować incydenty” (poziom operacyjny) oraz „oceniać poziom ryzyka wykrytych podatności” (poziom zarządczy/ekspercki).
- Pominięcie nowoczesnych technologii - mimo że podstawa programowa w innych miejscach wspomina o chmurze, w module testowania sieci (INF.11.7) całkowicie pominięto testowanie bezpieczeństwa infrastruktur hybrydowych i chmurowych, które są obecnie głównym wektorem ataków.

3. Metodyczna pułapka teoretyzacji i ryzyko „papierowego” egzaminu
Struktura kryteriów weryfikacji w jednostce INF.11.7 promuje postawę pasywną, co uniemożliwi rzetelną weryfikację umiejętności praktycznych na egzaminie zawodowym.
- Dominacja opisowości - kryteria takie jak: „wyjaśnia wpływ zagrożeń”, „identyfikuje niebezpieczne konfiguracje na podstawie zaleceń”, „opisuje znaczenie logów” czy „przygotowuje informacje do eskalacji” mają charakter czysto teoretyczny.
- Dewaluacja modelu wykonawczego - brak jednoznacznych kryteriów wymuszających fizyczne przeprowadzenie ataku lub obrony (np. „skutecznie blokuje atak DoS w konfiguracji firewall”) zmusza Centralną Komisję Egzaminacyjną do stworzenia zadań w modelu dokumentacyjnym (D).
- Konsekwencje - egzamin będzie sprawdzał umiejętność wypełniania raportu z testu, a nie umiejętność jego przeprowadzenia, co czyni dyplom państwowy bezwartościowym dla pracodawców z branży SOC (Security Operations Center).

4. Rażąca sprzeczność między wymogami technologicznymi a finansowaniem (OSR)
Jednostka INF.11.7 wymaga od szkół wdrożenia narzędzi klasy korporacyjnej, których koszt zakupu, licencji i utrzymania jest drastycznie zaniżony lub pominięty w Ocenie Skutków Regulacji.
- Wymagania infrastrukturalne: Zgodnie z „Warunkami realizacji kształcenia”, szkoła musi zapewnić m.in. oprogramowanie typu SIEM (Security Information and Event Management), systemy wykrywania i zapobiegania włamaniom IDS/IPS oraz profesjonalne skanery podatności (np. OpenVAS).

- Koszty sprzętowe: Prawidłowe działanie systemów SIEM i IDS/IPS wymaga wysokowydajnych serwerów oraz macierzy dyskowych do przechowywania i korelacji ogromnych ilości logów sieciowych.
- Błąd w OSR: Legislatorzy deklarują brak skutków dla budżetu państwa, podczas gdy stworzenie profesjonalnego laboratorium „Blue Team” (obrona i monitorowanie) to koszt rzędu setek tysięcy złotych na jedną placówkę.

4.7.2. Uwagi dotyczące wybranych efektów kształcenia INF.11.7

Efekt kształcenia	**Kryterium weryfikacji**	**Uwagi**
1) rozpoznaje zagrożenia bezpieczeństwa występujące w sieciach TCP/IP	1) wymienia podstawowe zagrożenia sieciowe, w tym:	Sformułowanie "w tym:" sugeruje że katalog nie jest zamknięty, z jednej strony autor wylicza zagrożenia ukierunkowując, czego będziemy uczyć i wymagać a z drugiej zostawia furtkę na bliżej nieznane
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	1) wykorzystuje narzędzia do identyfikacji ekspozycji i podatności sieciowej (nmap, skaner podatności):	Ten EK jest zbyt ambitny operacyjnie: wymaga od uczniów dokładnej znajomości mechanizów i ich kontroli krzyżowej, aby wychwytywać usterki i podatności. Realnie jest do zrealizowania tylko powierzchownie.
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	1) wykorzystuje narzędzia do identyfikacji ekspozycji i podatności sieciowej (nmap, skaner podatności): (...) a) skanuje sieci w celu wykrycia aktywnych hostów i otwartych portów z wykorzystaniem narzędzi typu nmap	Nazwa narzędzia jest już w 1)
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	1) wykorzystuje narzędzia do identyfikacji ekspozycji i podatności sieciowej (nmap, skaner podatności):	Nazwa narzędzia jest już w 1) Co uczniom da wykrycie podatności poza powiedzieniem, że "coś

	(...) c) wykonuje automatyczne skanowanie podatności z wykorzystaniem skanera podatności	się pokazało"?
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	2) sprawdza konfigurację VLAN, ACL i NAT pod kątem bezpieczeństwa: (...) c) blokowania sfałszowanych adresów przez ACL	Czy w tym punkcie nie ograniczamy funkcjonalnie ACL? Jakiś przypadek wyrwany z szerszej funkcjonalności, dlaczego właściwie ten?
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	3) identyfikuje niebezpieczne konfiguracje sieciowe na podstawie (np. zaleceń producentów, obowiązujących standardów oraz wewnętrznych polityk bezpieczeństwa)	Językowo do poprawienia. Propozycja zapisu: 3) identyfikuje niebezpieczne elementy konfiguracji bezpieczeństwa, porównując je w szczególności do zaleceń producentów oraz obowiązujących standardów i wewnętrznych polityk bezpieczeństwa Przy czym to też może być zbyt ambitne w tym brzmieniu - technik nie ma wyłapać niuansów, tylko typowe "mamy konto admin/admin" czy inne, podstawowe elementy konfiguracji.
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	4) ocenia skuteczność zastosowanych zabezpieczeń sieciowych:	Technik ocenia skuteczność ? To jest rola inżyniera bezpieczeństwa, audytora zewnętrznego. To nieporozumienie.
2) weryfikuje konfigurację sieci pod kątem bezpieczeństwa	4) wskazuje elementy konfiguracji wymagające poprawy	Błąd numeracji - powinno być 5

4) rozpoznaje i analizuje próby naruszenia bezpieczeństwa sieci TCP/IP	1) wyjaśnia znaczenie logów systemowych i sieciowych w procesie wykrywania i analizy prób naruszenia bezpieczeństwa	EK merytorycznie potrzebny jako wstępna analiza SOC, ale zbyt ambitny dydaktycznie dla uczniów klas 1-3 technikum (to jest pierwsza kwalifikacja, uczniowie mają 14-16 lat!).
4) rozpoznaje i analizuje próby naruszenia bezpieczeństwa sieci TCP/IP	3) rozpoznaje próby nieautoryzowanego dostępu na podstawie podstawowych informacji o zdarzeniach sieciowych:	Po co jest to wymienienie szczegółów? Każde narzędzie je wyświetla w ten lub inny sposób, zapisy tylko zaciemniają treść. Proponuję je usunąć.
4) rozpoznaje i analizuje próby naruszenia bezpieczeństwa sieci TCP/IP	4) rozpoznaje nietypowe wzorce ruchu i zdarzeń sieciowych mogące świadczyć o naruszeniu bezpieczeństwa, w tym:	To wydaje się już robota dla SIEM / IDS / IPS, nie dla technika. Nawet jeśli, to enumeracja zaczynająca się od "w tym" wskazuje przykłady, a nie pełny zakres. W efekcie można tu podpiąć cokolwiek. Proponuje się zapis: 4) rozpoznaje najczęstsze nietypowe wzorce ruchu i zdarzeń sieciowych mogące świadczyć o naruszeniu bezpieczeństwa, w szczególności: a) nagłego wzrostu liczby połączeń lub przesyłanych pakietów, b) ruchu sieciowego generowanego poza standardowymi godzinami pracy, c) komunikacji z nieznanymi lub podejrzanymi adresami IP i domenami.
5) analizuje incydenty sieciowe oraz opracowuje dokumentację analityczną	1) dokumentuje wykryte incydenty bezpieczeństwa	Proponuje się dopisek: "w formie krótkiego zgłoszenia lub raportu" -

cyberbezpieczeństwa		nie jest zadaniem technika przygotowywanie pełnego "post-mortem". Nie można tutaj mówić o pełnym formalizmie dokumentacji.
5) analizuje incydenty sieciowe oraz opracowuje dokumentację analityczną cyberbezpieczeństwa	2) przygotowuje informacje do eskalacji incydentu zgodnie z procedurami	To jest zakres na poziomie PRK 6 albo i 7 - nie jest adekwatny do deklarowanego poziomu PRK 4.

Rekomendacje:

1) Operacjonalizacja kryteriów - zmiana czasowników z „opisuje/wyjaśnia” na „przeprowadza skanowanie”, „wykrywa anomalię w ruchu”, „wdraża regułę IDS”, aby umożliwić egzamin zawodowy w modelu wykonawczym (Model W).
2) Urealnienie OSR - wskazanie, że realizacja jednostki INF.11.7 wymaga centralnych dotacji na licencje dla systemów SIEM oraz wysokowydajny sprzęt serwerowy do analizy Big Data (logi sieciowe).
3) Wydłużenie cyklu kształcenia - przesunięcie zagadnień z zakresu SIEM i reagowania na incydenty do klasy programowo najwyższej i zwiększenie liczby godzin, aby uniknąć kształcenia „teoretyków cyberbezpieczeństwa”.
4) Zgodność z certyfikacjami rynkowymi - postulat dostosowania zakresu testowania do standardów uznawanych przez przemysł (np. CompTIA Security+ lub licencjonowane ścieżki Blue Team), co podniosłoby prestiż dyplomu technika.

4.8. Dotyczy INF.12.3. Implementacja aplikacji webowych

4.8.1. Analiza jednostki INF.12.3

1. Merytoryczny anachronizm i brak neutralności technologicznej

Projekt jednostki wykazuje tendencję do faworyzowania technologii schodzących lub specyficznych rozwiązań kosztem nowoczesnych standardów rynkowych:

- Nacisk na technologie monolityczne - projekt podstawy programowej kładzie duży nacisk na tradycyjne podejście serwerowe (często utożsamiane w praktyce szkolnej z PHP), marginalizując nowoczesne architektury typu SPA (Single Page Application), Microservices czy systemy oparte na Node.js/TypeScript, które są obecnie standardem w branży cyberbezpieczeństwa i systemach wysokiej dostępności.

2. Błąd metodyczny: Nadużywanie AI w procesie nauki składni
Podobnie jak w innych modułach programistycznych, jednostka INF.12.3 wprowadza wymóg stosowania AI do generowania kodu frontendowego (HTML/CSS) i backendowego.

- Utrata fundamentów - na etapie nauki implementacji aplikacji webowej, uczeń musi samodzielnie opanować logikę modelu DOM oraz asynchroniczność (Promises/Async-Await). Automatyzacja tego procesu przez AI na poziomie 90-godzinnego lub 150-godzinnego kursu doprowadzi do powstania „programistów kopiuj-wklej", którzy nie potrafią debugować wygenerowanego kodu, gdy AI popełni błąd logiczny lub wprowadzi lukę bezpieczeństwa.
- Problem z walidacją - kryterium „weryfikuje kod z AI" jest iluzoryczne, jeśli uczeń nie posiada ugruntowanej wiedzy o standardach ECMAScript.

3. Dydaktyczne przeładowanie: Od layoutu po API w 150 godzinach
Zakres materiału przewidziany dla jednostki INF.12.3 jest niemożliwy do rzetelnego opanowania w założonym czasie, co narusza zasady gradacji wiedzy (PRK):

- Zbyt szeroki stos technologiczny - w jednym module uczeń ma opanować: zaawansowany CSS (Flexbox/Grid), preprocesory (SASS/LESS), język JavaScript (ES6+), frameworki frontendowe (np. React/Angular), technologie backendowe, obsługę baz danych, integrację z zewnętrznymi API oraz testowanie (Unit Tests/E2E).
- Powierzchowność kształcenia - tak szerokie spektrum wymusza na nauczycielu „przebiegnięcie" przez tematy, co uniemożliwia uczniowi zrozumienie np. cyklu życia komponentu czy mechanizmów zabezpieczania sesji (JWT, OAuth2).

4. Metodyczne ryzyko teoretyzacji (Model D)
Mimo że nazwa jednostki to „Implementacja", co sugeruje tworzenie kodu, kryteria weryfikacji ponownie uciekają w stronę wiedzy opisowej:

- Pasywne kryteria - uczeń ma: „wyjaśniać zasady responsywności", „charakteryzować metody przesyłania danych", „opisywać protokół HTTP".
- Zagrożenie egzaminacyjne - taka konstrukcja kryteriów pozwala na przeprowadzenie egzaminu zawodowego w formie testu lub prostego opisu struktury aplikacji (Model D), zamiast fizycznego zakodowania funkcjonalnego i bezpiecznego modułu aplikacji (Model W). Absolwent z dyplomem może nie potrafić samodzielnie uruchomić prostego serwera API.

5. Ignorowanie kosztów infrastruktury i narzędzi (Błąd w OSR)
Realizacja nowoczesnych aplikacji webowych wymaga odpowiedniego środowiska, co zostało pominięte w analizie finansowej projektu:

- Wymagane zasoby: Środowiska do konteneryzacji (Docker), płatne lub limitowane dostępy do narzędzi typu Cloud IDE, systemy CI/CD (Continuous Integration/Deployment) oraz serwery testowe.

- Sprzeczność z OSR: Autorzy projektu zakładają brak kosztów, podczas gdy profesjonalne nauczanie web-devu wymaga nowoczesnych stacji roboczych zdolnych do płynnej pracy z ciężkimi środowiskami programistycznymi (IDE) oraz lokalnymi instancjami baz danych i serwerów aplikacji.

4.8.2. Uwagi dotyczące wybranych efektów kształcenia INF.12.3

Efekt kształcenia	**Kryterium weryfikacji**	**Uwagi**
1) programuje obiektowo w języku Java (lub C#)	1) stosuje narzędzia programisty do pracy własnej i grupowej: IDE, GIT	W jakim celu uczeń wykorzystuje język Java w cyberbezpieczeństwie?
1) programuje obiektowo w języku Java (lub C#)	6) wykorzystuje podstawowe techniki programowania współbieżnego: wątki oraz mechanizmy synchronizacji (dla C# również mechanizm async/await): (...) a) uruchamiając i zatrzymując wątki w sposób kontrolowany	Niepoprawne formy językowe. Należy użyć czasowników w 2. osobie liczby pojedynczej, nie w imiesłowie odczasownikowym. Dotyczy całego 6) KW. a) uruchamia i zatrzymuje wątki w sposób kontrolowany, b) wykorzystuje pule wątków, c) synchronizuje wątki z wykorzystaniem semaforów lub barier, d) komunikuje się z wątkami za pomocą kolejek, e) synchronizuje dostęp do danych.
3) programuje aplikacje webowe	1) wyjaśnia i stosuje technologie klienta: HTML, CSS, JavaScript	Przecież to jest INF.03 ściśnięte do kilku KW (1-3). Gdzie stawiamy granicę pomiędzy informatykiem/programistą a cyberbezpieczeństwem w tym zakresie? To KW pokrywa całą kwalifikację INF.03!!!
4) konfiguruje oraz uruchamia aplikacje webowe	1) bezpiecznie konfiguruje aplikację webową i web serwer (lub proxy serwer):	Brak uczniowi wiedzy na temat przygotowania serwera, a w tym

		kryterium uczeń ma web serwer konfigurować.
4) konfiguruje oraz uruchamia aplikacje webowe	1) bezpiecznie konfiguruje aplikację webową i web serwer (lub proxy serwer): (...) a) wymusza stosowanie protokołu HTTPS oraz kluczowych nagłówków	A gdzie wiedza na temat przygotowania certyfikatu. Uczeń nie posiada wiedzy na ten temat.
4) konfiguruje oraz uruchamia aplikacje webowe	4) stosuje podstawowe mechanizmy bezpieczeństwa kontenera Docker:	Brak wprowadzenia technologii Docerowej.

Rekomendacje:

1) Operacjonalizacja na rzecz bezpieczeństwa - zmiana kryteriów na wykonawcze, np.: „implementuje mechanizmy ochrony przed CSRF”, „waliduje dane wejściowe po stronie serwera zgodnie z OWASP”, „konfiguruje bezpieczne nagłówki HTTP (HSTS, CSP)”.
2) Podział na frontend i backend - sugeruje się wydzielenie technologii frontendowych i backendowych do osobnych jednostek, aby zapewnić uczniom czas na zrozumienie logiki każdej z warstw aplikacji.
3) Neutralność technologiczna - usunięcie sugestii konkretnych bibliotek na rzecz wymagań funkcjonalnych (np. „wykorzystuje framework klasy SPA” zamiast wskazywania konkretnej nazwy, która za 2 lata może być nieaktualna).
4) Urealnienie roli AI - narzędzia AI powinny być dopuszczone jedynie w fazie refaktoryzacji i optymalizacji gotowego, napisanego samodzielnie przez ucznia kodu, a nie jako źródło generowania logiki biznesowej aplikacji.

4.9. Dotyczy INF.12.4. Weryfikowanie bezpieczeństwa implementacji aplikacji webowych

1. Błąd w profilowaniu zawodowym: programista vs. specjalista w zakresie cyberbezpieczeństwa

Analiza treści jednostki INF.12.4 wskazuje na fundamentalne niezrozumienie ról zawodowych przez autorów projektu.

- Kompetencje deweloperskie - jednostka skupia się niemal wyłącznie na „weryfikacji implementacji”, co w przemyśle jest domeną Senior Developerów lub QA Automation Engineerów (testerów oprogramowania). Technik cyberbezpieczeństwa nie jest „poprawiaczem kodu” – jego rolą jest audytowanie systemów, wykrywanie wektorów ataku i zabezpieczanie infrastruktury, na której aplikacja pracuje.

- Brak perspektywy infrastrukturalnej - w module całkowicie pominięto aspekty kluczowe dla cyberbezpieczeństwa, takie jak: konfiguracja WAF (Web Application Firewall), zabezpieczanie serwerów brzegowych (Reverse Proxy), zarządzanie certyfikatami SSL/TLS na poziomie serwera czy ochrona przed atakami DDoS. Skupienie się na „sprawdzaniu poprawności kodu" przesuwa ten zawód niebezpiecznie w stronę technika programisty, dublując kompetencje i osłabiając profil specjalisty ds. bezpieczeństwa.

2. Metodyczna pułapka teoretyzacji (Model D)
Mimo że nazwa jednostki sugeruje działania audytowe, kryteria weryfikacji są sformułowane w sposób, który promuje wiedzę encyklopedyczną kosztem praktycznych umiejętności „offensywnych" i „defensywnych":

- Czasowniki pasywne - uczeń ma: „wyjaśniać pojęcia", „charakteryzować podatności", „opisywać techniki", „wymieniać narzędzia".
- Fikcja audytu - rakuje wymogu przeprowadzenia realnego testu penetracyjnego zakończonego skutecznym wykryciem i eksploatacją luki (w kontrolowanym środowisku). Zamiast tego uczeń ma „opisywać znaczenie raportu".
- Skutek dla egzaminu - taka konstrukcja wymusza egzamin w modelu dokumentacyjnym (D). Absolwent będzie potrafił nazwać podatność z listy OWASP Top 10, ale nie będzie umiał jej samodzielnie zidentyfikować przy użyciu profesjonalnych narzędzi (np. Burp Suite czy OWASP ZAP).

3. Ryzyko bezkrytycznego polegania na AI
Efekt kształcenia nr 4 wprowadza AI jako narzędzie do weryfikacji bezpieczeństwa kodu i generowania scenariuszy testowych.

- Zagrożenie „False Sense of Security": Uczeń na poziomie technikum nie posiada wystarczającej wiedzy merytorycznej, by zweryfikować, czy AI nie pominęła krytycznej luki (False Negative) lub nie wskazała błędu tam, gdzie go nie ma (False Positive).
- Dydaktyczny regres: Narzędzia AI powinny być nauczane jako wsparcie dla eksperta, a nie jako fundament dla nowicjusza. Wprowadzenie ich na tym etapie zdejmuje z ucznia obowiązek zrozumienia niskopoziomowej logiki protokołu HTTP i mechanizmów sesji.

4. Dydaktyczne przeładowanie i brak gradacji (PRK)
Jednostka próbuje w minimalnym wymiarze godzin (zazwyczaj ok. 90h) zamknąć potężny obszar wiedzy, jakim jest Web Pentesting:

- Nierealny zakres - od statycznej analizy kodu (SAST), przez dynamiczną (DAST), po testy fuzingu, audytowanie API (REST/GraphQL) i analizę logów serwerowych. Każdy z tych elementów w przemyśle jest osobną ścieżką certyfikacji.
- Chaos poziomów - wymaganie od ucznia technikum „weryfikacji mechanizmów kryptograficznych aplikacji" przy jednoczesnym braku

głębokich podstaw matematycznych i programistycznych jest dydaktycznie niemożliwe do zrealizowania inaczej niż poprzez powierzchowne „klikanie w narzędzia” bez zrozumienia ich działania.

5. Ukryte koszty i błędy w OSR (Infrastruktura laboratoriów)
Realizacja jednostki INF.12.4 wymaga specyficznego i kosztownego środowiska, o którym OSR milczy:

- Środowisko „Vulnerable by Design” - szkoła musi dysponować izolowaną infrastrukturą (np. lokalne instancje DVWA, OWASP Juice Shop), aby uczniowie mogli bezpiecznie przeprowadzać testy ataków bez naruszania prawa i bezpieczeństwa sieci szkolnej.
- Koszty licencji - istnieją narzędzia open-source, ale profesjonalne kształcenie wymaga kontaktu z narzędziami standardowymi dla branży (np. Burp Suite Professional), których koszt subskrypcji jest całkowicie pominięty w budżecie reformy.

Uwaga krytyczna: ta jednostka efektów kształcenia została przygotowana poprzez proste skopiowanie umiejętności programistów, zamiast zbudować unikalny profil specjalisty od ochrony infrastruktury krytycznej.

Rekomendacje:

1) Zmiana profilu jednostki - przekształcenie modułu z „Weryfikacji implementacji” (zadanie programisty) na „Testowanie bezpieczeństwa usług webowych” (zadanie specjalisty cyberbezpieczeństwa), kładąc nacisk na infrastrukturę i protokoły, a nie na analizę linijek kodu źródłowego.
2) Operacjonalizacja kryteriów - zmiana czasowników na „przeprowadza skanowanie”, „wykrywa podatność”, „dokumentuje dowód ataku (PoC)”, co pozwoli na egzamin zawodowy w modelu wykonawczym (Model W).
3) Urealnienie OSR - wskazanie konieczności finansowania izolowanych laboratoriów serwerowych (typu Cyber Range) niezbędnych do bezpiecznej nauki testów penetracyjnych.
4) Usunięcie dublowania kompetencji - przeniesienie głębokiej analizy kodu (SAST) do podstawy programowej dla zawodu technik programista, pozostawiając technikowi cyberbezpieczeństwa analizę dynamiczną i konfigurację zabezpieczeń sieciowych aplikacji.

4.10. Dotyczy INF.12.5. Przeprowadzanie testów penetracyjnych aplikacji webowych w środowisku testowym

1. Błąd merytoryczny: Niedopasowanie poziomu PRK do stopnia złożoności (Script Kiddie vs. Pentester)
Jednostka INF.12.5 nakłada na ucznia technikum zadania, które w sektorze komercyjnym wymagają poziomu inżynierskiego lub specjalistycznych certyfikacji (np. OSCP, eWPT).

- Powierzchowność kształcenia - w ramach zaledwie 90 godzin uczeń ma opanować techniki ataku na bazy danych (SQL Injection), przejmowanie sesji, ataki XSS, CSRF oraz omijanie mechanizmów uwierzytelniania.
- Ryzyko dydaktyczne - tak krótki czas przy tak szerokim materiale promuje postawę tzw. „Script Kiddie” – bezmyślnego uruchamiania gotowych skryptów i narzędzi (np. SQLmap) bez głębokiego zrozumienia protokołu HTTP, logiki aplikacji i mechanizmów obronnych. Jest to zaprzeczenie idei kształcenia specjalisty ds. bezpieczeństwa.

2. Krytyczne braki w zakresie bezpieczeństwa prawnego i etyki
Ofensywne działania w sieci (testy penetracyjne) wiążą się z ogromną odpowiedzialnością prawną (art. 267–269b Kodeksu Karnego).

- Brak procedur - podstawa programowa wspomina o „etyce”, ale całkowicie pomija naukę procedur formalno-prawnych niezbędnych w zawodzie, takich jak przygotowanie Rules of Engagement (RoE) czy uzyskiwanie pisemnej zgody na testy.
- Zagrożenie dla szkół - wypuszczenie uczniów z wiedzą o narzędziach ataku, bez ugruntowanych procedur prawnych i izolowanych laboratoriów, naraża placówki oświatowe oraz samych uczniów na konflikt z prawem w przypadku prób „testowania” systemów poza szkołą.

3. Fikcja infrastrukturalna i finansowa (Błąd w OSR)
Realizacja jednostki INF.12.5 w sposób praktyczny wymaga stworzenia zaawansowanych, izolowanych środowisk typu Cyber Range lub Vulnerable-by-Design.

- Koszty sprzętu i izolacji - przeprowadzanie testów penetracyjnych nie może odbywać się w standardowej sieci szkolnej. Wymagane są dedykowane serwery do hostowania podatnych aplikacji oraz stacje robocze z systemami typu Kali Linux, odizolowane fizycznie lub logicznie (VLAN/VPN).
- Licencje - profesjonalne testy wymagają narzędzi klasy Burp Suite Professional czy komercyjnych skanerów. Ich koszt jest całkowicie pominięty w OSR, co zmusza szkoły do korzystania z wersji darmowych (Community), które nie odzwierciedlają standardów pracy w nowoczesnych firmach cybersecurity.

4. Metodyczna teoretyzacja: „Papierowy” atak (Model D)
Kryteria weryfikacji w INF.12.5 ponownie wykazują skłonność do teoretyzowania umiejętności, które z natury są czysto praktyczne:
Pasywne kryteria: Uczeń ma: „wyjaśniać podatności”, „charakteryzować typy ataków”, „opisywać wektory”, „wskazywać znaczenie raportów”.

- Brakuje kluczowego kryterium wykonawczego: „skutecznie identyfikuje i eksploatuje lukę w zabezpieczeniach w środowisku testowym oraz proponuje skuteczną metodę jej naprawy (remediacji)”.
- Egzamin zawodowy w obecnej formie (Model D – dokumentacyjny) z „testów penetracyjnych” będzie polegał na opisie ataku na kartce papieru lub w arkuszu kalkulacyjnym. Taka forma walidacji kompetencji jest w branży IT traktowana jako bezwartościowa.

5. Przeładowanie i chaos dydaktyczny
Jednostka próbuje wtłoczyć w 90 godzin materiał, który w profesjonalnych kursach zajmuje setki godzin intensywnych laboratoriów.

- Chaos merytoryczny - w jednym module uczeń ma opanować: rekonesans (OSINT), skanowanie, ataki webowe (OWASP Top 10), testy API, ataki na infrastrukturę chmurową oraz raportowanie.
- Nierealność celów - przy 90 godzinach (3 godziny tygodniowo) uczeń po przerwie wakacyjnej lub świątecznej nie będzie w stanie utrzymać ciągłości technicznej niezbędnej do zrozumienia złożonych łańcuchów ataku (Exploit Chains).

Uwaga krytyczna:
Warto zaznaczyć, że jednostki od INF.12.3 do INF.12.5 tworzą ciąg logiczny, który w obecnym projekcie jest wadliwy na każdym etapie: od błędnej implementacji, przez powierzchowną weryfikację kodu, aż po teoretyczne testy penetracyjne. To prosta droga do wykształcenia absolwentów, którzy posiadają niebezpieczną „iluzję kompetencji”, niepopartą rzetelną praktyką i zrozumieniem fundamentów bezpieczeństwa.

Rekomendacje:

1) Operacjonalizacja kryteriów - zmiana czasowników na „przeprowadza”, „eksploatuje”, „naprawia”, „audytuje”. Egzamin musi mieć formę praktycznego wyzwania typu CTF (Capture The Flag) w izolowanym środowisku.
2) Urealnienie OSR - wskazanie na konieczność finansowania licencji na oprogramowanie specjalistyczne oraz dedykowanych serwerów do wirtualizacji środowisk podatnych.
3) Wzmocnienie modułu prawnego - wprowadzenie obowiązkowych efektów kształcenia dotyczących aspektów prawnych przeprowadzania testów (umowy, zgody, odpowiedzialność karna).
4) Zwiększenie wymiaru godzinowego - pentesting aplikacji webowych to najbardziej złożona część zawodu. Wymaga ona co najmniej 150–180 godzin realnej pracy laboratoryjnej, aby wyjść poza poziom „opisywania obrazków”.

4.11. Dotyczy INF.11.8. i INF.12.6. Język angielski zawodowy

1. Strategiczny błąd merytoryczny: Angielski jako „dodatek”, a nie fundament

W branży cyberbezpieczeństwa 100% dokumentacji technicznej (RFC, normy ISO/IEC, opisy podatności CVE, bazy MITRE ATT&CK) oraz większość narzędzi (CLI, Wireshark, systemy SIEM) operuje w języku angielskim.

- Niewystarczający wymiar godzin - przeznaczenie zaledwie 30 godzin na język angielski zawodowy dla technika cyberbezpieczeństwa jest rażącym niedoszacowaniem. W tym czasie uczeń nie jest w stanie opanować specjalistycznej terminologii z zakresu kryptografii, architektury sieciowej i inżynierii wstecznej.
- Brak integracji z przedmiotami zawodowymi - podstawa programowa traktuje język obcy jako odizolowaną wyspę, zamiast wymusić korzystanie z anglojęzycznych instrukcji i dokumentacji bezpośrednio na laboratoriach z konfiguracji serwerów czy testów penetracyjnych.

2. Metodyczna teoretyzacja komunikacji (Model D)
Kryteria weryfikacji w obu jednostkach są sformułowane w sposób, który promuje bierną znajomość języka, zamiast biegłości operacyjnej:

- Pasywne kryteria - uczeń ma: „rozróżniać terminy", „wyjaśniać znaczenie pojęć", „analizować teksty".
- Brak kompetencji produktywnych - w branży cyberbezpieczeństwa kluczowe jest sporządzanie raportów z incydentów (Incident Reports), pisanie dokumentacji powykonawczej oraz komunikacja w międzynarodowym zespole (np. poprzez systemy biletowe typu Jira/ServiceNow). Te aspekty są w podstawie potraktowane marginalnie.
- Problem egzaminacyjny - egzamin zawodowy sprawdzający język angielski w formie testu pisemnego (fragment zrzutu ekranu czy zdjęcia z komunikatem w języku angielskim) nie weryfikuje zdolności czytania ze zrozumieniem krytycznych alertów systemowych w czasie rzeczywistym.

3. Błąd metodyczny - dublowanie treści i brak profilowania
Jednostki INF.11.8 i INF.12.6 są w dużej mierze identyczne, co prowadzi do nieefektywności dydaktycznej:

- Brak specjalizacji - język angielski dla administratora sieci (INF.11) powinien skupiać się na terminologii hardware'owej, protokołach i topologiach, natomiast dla (INF.12) na składni, logice algorytmów i dokumentacji API.

4. Ignorowanie roli AI i tłumaczenia maszynowego w 2026 roku
W dobie zaawansowanych modeli językowych (LLM), nauka języka zawodowego musi ulec transformacji, czego projekt rozporządzenia nie zauważa:

- Brak krytycznego myślenia - podstawa nie zawiera efektów kształcenia dotyczących weryfikacji technicznej tłumaczeń generowanych przez AI. W

cyberbezpieczeństwie błędne przetłumaczenie jednego parametru w skrypcie lub procedurze hardeningu może doprowadzić do krytycznej awarii systemu.
- Anachronizm dydaktyczny - uczenie się na pamięć słówek w 2026 roku, bez umiejętności sprawnego korzystania z narzędzi wspomagających tłumaczenie dokumentacji technicznej i ich merytorycznej korekty, jest stratą czasu dydaktycznego.

5. Bariera kadrowa i metodyczna

Realizacja tych jednostek napotyka potężny problem ludzki:

- Kompetencje nauczycieli - języka angielskiego zawodowego często uczą filolodzy bez przygotowania technicznego. Nie są oni w stanie wyjaśnić uczniowi niuansów między „authentication" a „authorization" czy specyfiki terminologii związanej z „payloadem" w ataku, co prowadzi do nauczania pustych definicji.
- Brak materiałów - ministerstwo nie wskazuje źródeł ani baz wiedzy (np. anglojęzycznych forów technicznych), na których uczeń mógłby trenować język w naturalnym środowisku pracy.

Rekomendacje:

1) Integracja języka z praktyką - postulat, aby język angielski zawodowy był realizowany metodą CLIL (Content and Language Integrated Learning) bezpośrednio przez nauczycieli przedmiotów zawodowych lub we współpracy z anglistą podczas laboratoriów technicznych. W branży IT sprzyja temu pełna dokumentacja w języku angielskim czy materiały dydaktyczne producenta sprzętu.
2) Operacjonalizacja na rzecz dokumentacji - zmiana kryteriów na: „przygotowuje raport z incydentu w j. angielskim", „analizuje opis podatności w bazie CVE", „prowadzi komunikację techniczną w systemie zgłoszeniowym".
3) Wprowadzenie krytyki AI - dodanie efektów kształcenia dotyczących poprawnej weryfikacji dokumentacji technicznej tłumaczonej automatycznie.
4) Urealnienie wymiaru godzin - zwiększenie liczby godzin języka zawodowego kosztem teoretycznych wstępów w innych modułach, uznając go za kluczową kompetencję „twardą" w cyberbezpieczeństwie.
5) Wymaganie dodatkowe godziny - nauczyciele wszystkich obowiązkowych zajęć edukacyjnych z zakresu kształcenia zawodowego powinni stwarzać uczniom warunki do nabywania kompetencji personalnych i społecznych. Nie jest to tylko metoda pracy, bo wymaga wprowadzenia merytorycznego w zagadnienia w związku z tym wymaga przeznaczenia określonej minimalnej liczby godzin.

4.12. Dotyczy INF.11.9. i INF.12.7. Kompetencje personalne i społeczne

1. Błąd merytoryczny - brak profilowania pod kątem „Just Culture” i etyki hakerskiej

W cyberbezpieczeństwie kompetencje społeczne nie są „miękkim dodatkiem”, lecz fundamentem bezpieczeństwa organizacji. Projekt rozporządzenia całkowicie to pomija:

- Etyka zawodowa vs. Etyka hakerska - podstawa posługuje się ogólnikami o „przestrzeganiu zasad etyki”, podczas gdy uczeń technikum cyberbezpieczeństwa musi zrozumieć różnicę między działaniem White Hat a Black Hat, wagę zachowania poufności (NDA) oraz etyczne aspekty ujawniania podatności (Responsible Disclosure).
- Brak kultury bezpieczeństwa (Just Culture) - w nowoczesnym IT kluczowa jest umiejętność przyznania się do błędu/luki bez strachu przed karą, co pozwala na szybką reakcję (Incident Response). Projekt skupia się na „planowaniu zadań”, a pomija psychologię pracy w sytuacjach kryzysowych i kulturę raportowania błędów własnych.

2. Metodyczna pułapka niemierzalności (Fikcja oceny)

Jednostki KPS są sformułowane przy użyciu czasowników, których nie da się rzetelnie zweryfikować podczas egzaminu zawodowego ani rzetelnie ocenić w procesie dydaktycznym:

- Niemierzalne kryteria - jak obiektywnie ocenić, czy uczeń „jest otwarty na zmiany”, „wykazuje kreatywność” lub „przestrzega zasad kultury osobistej”?
- Problem egzaminacyjny - ponieważ kompetencji tych nie da się sprawdzić w modelu wykonawczym (W), CKE zmuszona jest do zadawania pytań testowych typu: „Co zrobisz, gdy zobaczysz błąd kolegi?”. Uczeń wybiera „poprawną politycznie” odpowiedź, co w żaden sposób nie weryfikuje jego realnych postaw społecznych czy etycznuych. Jest to klasyczny przykład Modelu D (dokumentacyjnego) w obszarze, który wymaga modelu behawioralnego.

3. Dydaktyczny anachronizm - ignorowanie inżynierii społecznej (Social Engineering)

W zawodzie technika cyberbezpieczeństwa kompetencje społeczne są najczęstszym wektorem ataku.

- Brak perspektywy obronnej - nauczanie komunikacji społecznej (KPS) powinno tu być nierozerwalnie połączone z nauką rozpoznawania manipulacji, technik wywierania wpływu i ochrony przed socjotechniką (phishing, vishing, pretexting).

- Błąd logiczny - projekt traktuje KPS jako narzędzie do „miłej współpracy w zespole”, całkowicie ignorując fakt, że dla specjalisty bezpieczeństwa „kompetencja społeczna” to także asertywność w odmawianiu dostępu i rygorystyczne przestrzeganie procedur, nawet pod presją autorytetu czy czasu.

4. Systemowa nieefektywność - dublowanie bloków (Copy-Paste)
Jednostki INF.11.9 i INF.12.7 są identyczne i stanowią „standardowy wsad” stosowany w niemal każdym zawodzie technikalnym:

- Brak gradacji PRK - te same kompetencje przypisuje się mechanikowi, kucharzowi i technikowi cyberbezpieczeństwa. Jednak odpowiedzialność społeczna osoby zarządzającej infrastrukturą krytyczną państwa (KSC) jest nieporównywalnie większa i wymaga zupełnie innego poziomu dojrzałości zawodowej, czego podstawa nie różnicuje.
- Marnotrawstwo czasu - przeznaczenie ok. 30-40 godzin na powtarzanie ogólników o „dbaniu o czystość na stanowisku pracy” w zawodzie, w którym brakuje godzin na realne laboratoria z kryptografii czy testów penetracyjnych, jest błędem.

5. Ignorowanie wpływu AI na relacje zawodowe
W 2026 roku kompetencje społeczne obejmują także współpracę człowieka z systemami AI (Human-AI Teaming).

- Brak kompetencji przyszłości - projekt milczy o etyce korzystania z AI w komunikacji zawodowej, o odpowiedzialności za błędy wygenerowane przez maszyny oraz o zagrożeniach dla prywatności wynikających z interakcji z botami.

Uwaga krytyczna:
Analiza jednostek KPS dopełnia obrazu projektu jako niespójnej kompilacji. Zamiast kształcić odpowiedzialnego, asertywnego i świadomego zagrożeń etycznych specjalistę, ministerstwo proponuje „standardowy moduł grzecznościowy”, który w żaden sposób nie przygotowuje do trudnych realiów pracy w sektorze bezpieczeństwa narodowego i infrastruktury krytycznej.

Rekomendacje
1) Profilowanie branżowe - postulat zastąpienia generycznego bloku KPS jednostką pt. „Etyka zawodowa i psychologia cyberbezpieczeństwa”, skupioną na odpowiedzialności prawnej, etyce hakerskiej i odporności na socjotechnikę.
2) Integracja z zadaniami zawodowymi - kompetencje personalne powinny być oceniane „przy okazji” laboratoriów technicznych (np. praca w zespole SOC przy obsłudze incydentu pod presją czasu), a nie jako teoretyczne wykłady o

„kreatywności”. Tutaj jednak brakuje kompetencji nauczycielom i jest ogromna potrzeba stosownych szkoleń doskonalących w tym zakresie.

3) Wprowadzenie mechanizmów Just Culture - nauka raportowania błędów i incydentów bez lęku przed sankcjami jako kluczowa kompetencja społeczna specjalisty IT.
4) Urealnienie kryteriów - usunięcie zapisów niemierzalnych (np. o „otwartości na zmiany”) na rzecz konkretnych postaw zawodowych (np. „przestrzega procedur dostępu”, „zachowuje poufność danych w symulowanym środowisku pracy”).

4.13. Dotyczy INF.11.10. i INF.12.8 Organizacja pracy małych zespołów

1. Błąd merytoryczny - anachronizm metodyczny wynikający z braku Agile/ Scrum/ DevSecOps

W roku 2026 standardem w branży IT jest praca w metodykach zwinnych. Projekt rozporządzenia zdaje się tego nie dostrzegać:

- Podejście hierarchiczne - podstawa promuje tradycyjny, liniowy model zarządzania („planowanie zadań”, „podział ról przez lidera”), który jest całkowicie nieprzystający do dynamiki pracy w cyberbezpieczeństwie.
- Brak kluczowych pojęć - w jednostkach tych nie pojawiają się pojęcia takie jak Scrum, Kanban, Sprint, Backlog czy Stand-up, które są elementarzem dla każdego pracownika sektora IT. Uczenie „organizacji pracy” bez odniesienia do tych narzędzi to kształcenie kadr do realiów biurowych z lat 90. ubiegłego wieku.

2. Ignorowanie specyfiki zespołów Cyberbezpieczeństwa (SOC/CSIRT)

Praca zespołowa w cyberbezpieczeństwie różni się od standardowego „zespołu projektowego”:

- Struktura Red/Blue/Purple Team - podstawa nie uwzględnia specyfiki współpracy między zespołami ofensywnymi a defensywnymi, co jest kluczowe dla efektywnego testowania bezpieczeństwa.
- Zarządzanie incydentem pod presją czasu - jednostka „Organizacja pracy” powinna kłaść nacisk na protokoły komunikacji kryzysowej i przekazywanie zadań w trybie 24/7 (Handover), a nie na ogólnikowe „dobieranie współpracowników do zadań”.

3. Metodyczna pułapka teoretyzacji i niemierzalności (fikcja egzaminacyjna)

Kryteria weryfikacji w tych jednostkach są niemożliwe do rzetelnej oceny w systemie egzaminów zawodowych:

- Niemierzalne kryteria - jak obiektywnie ocenić na egzaminie, czy uczeń „wykazuje dbałość o jakość pracy zespołu” lub „stosuje techniki motywacji” w sytuacji, gdy na egzaminie jest oceniany indywidualnie?
- Problem Modelu D - ponieważ nie da się tego sprawdzić w działaniu (bez długofalowej obserwacji pracy grupowej), egzamin zawodowy zostanie sprowadzony do teoretycznego testu wyboru: „Która cecha najlepiej opisuje lidera?”. Absolwent pozna definicję „lidera”, ale nie nabędzie umiejętności zarządzania zadaniami w systemie typu Jira czy Trello.

4. Błąd dydaktyczny - dublowanie i marnotrawstwo czasu (kopiuj i wklej)
Jednostki INF.11.10 i INF.12.8 są identyczne i stanowią kolejny „standardowy moduł”, który wklejono bez żadnej refleksji nad specyfiką zawodu:

- Brak gradacji PRK - te same efekty przypisuje się technikowi cyberbezpieczeństwa, co technikowi żywienia. Jednak odpowiedzialność za organizację pracy w zespole obsługującym infrastrukturę krytyczną wymaga zupełnie innych kompetencji (np. rygorystyczne trzymanie się procedur bezpieczeństwa, audytowalność decyzji).

5. Brak narzędzi IT w procesie organizacji
W 2026 roku organizacja pracy małych zespołów w IT odbywa się za pomocą zaawansowanego oprogramowania (Project Management Tools):

- Technologiczna pustka - podstawa nie wymusza nauki obsługi systemów ticketowych, tablic Kanbanowych (cyfrowych) czy narzędzi do automatyzacji workflow (np. CI/CD w kontekście bezpieczeństwa).
- Brak AI w zarządzaniu - projekt ignoruje fakt, że AI wspomaga obecnie planowanie zasobów, estymację czasu zadań i optymalizację pracy zespołów programistycznych.

Uwaga krytyczna
Organizacja pracy małych zespołów w kontekście zawodu technika cyberbezpieczeństwa jest jaskrawym przykładem kopiowania anachronicznych wzorców menedżerskich, które całkowicie ignorują nowoczesne metodyki pracy w IT (Agile, Scrum, DevSecOps) oraz specyfikę zespołów reagowania na incydenty (SOC/CSIRT).

Rekomendacje
1) Profilowanie pod Agile/Scrum - postulat zastąpienia generycznego bloku „Organizacja pracy” jednostką pt. „Metodyki pracy zwinnej w zespołach cyberbezpieczeństwa”.
2) Operacjonalizacja poprzez narzędzia - zmiana kryteriów na: „zarządza zadaniami w systemie klasy TMS/Jira”, „prowadzi dokumentację sprintu”, „planuje pracę zespołu zgodnie z metodyką Kanban”.

3) Wprowadzenie protokołów reagowania - nauczanie organizacji pracy w oparciu o standardy NIST czy SANS (Incident Handling Lifecycle) jako realnej formy współpracy zespołowej w cyberbezpieczeństwie.
4) Integracja z laboratoriami - organizacja pracy powinna być oceniana podczas wspólnych projektów technicznych (np. jeden uczeń konfiguruje serwer, drugi atakuje, trzeci monitoruje logi), a nie jako osobny przedmiot wykładowy.
5) Wymaganie dodatkowe godziny - nauczyciele wszystkich obowiązkowych zajęć edukacyjnych z zakresu kształcenia zawodowego powinni stwarzać uczniom warunki do nabywania organizacji pracy małych zespołów. Nie jest to tylko metoda pracy, bo wymaga wprowadzenia merytorycznego w zagadnienia w związku z tym wymaga przeznaczenia określonej minimalnej liczby godzin.

5. Miejsce odbywania praktyk zawodowych

5.1. Obecnie wskazane miejsca odbywania praktyk zawodowych

Analiza wykazu miejsc realizacji praktyk zawodowych w zestawieniu z rzeczywistością prawną i operacyjną sektora cyberbezpieczeństwa wskazuje na istotny brak realizmu w kontekście uczniów technikum (osób niepełnoletnich). Dokument wymienia m.in. jednostki samorządu terytorialnego, centra danych, operatorów infrastruktury krytycznej oraz przedsiębiorstwa zajmujące się obsługą systemów IT. Wielu wymienionych pracodawców, szczególnie operatorzy infrastruktury krytycznej oraz jednostki samorządu terytorialnego, operuje na systemach objętych ustawą o ochronie informacji niejawnych lub ustawą o Krajowym Systemie Cyberbezpieczeństwa. Osoby niepełnoletnie nie mogą przejść pełnej procedury sprawdzającej (ankiety bezpieczeństwa osobowego) w celu uzyskania poświadczenia bezpieczeństwa do klauzul "poufne" czy "zastrzeżone". Wpuszczenie praktykanta bez poświadczenia do pomieszczeń, gdzie znajdują się ekrany z danymi operacyjnymi (np. w SOC - Security Operations Center), jest naruszeniem procedur bezpieczeństwa fizycznego.
Praktyki zawodowe w zawodzie technik cyberbezpieczeństwa wiążą się z dostępem do wrażliwej infrastruktury (serwery, bazy danych, logi). Uczeń niepełnoletni nie może w pełni skutecznie podpisać umowy o odpowiedzialności materialnej za mienie wielkiej wartości ani rygorystycznych umów o poufności (NDA) bez zgody opiekuna prawnego, co dla wielu firm z sektora IT jest barierą biurokratyczną nie do przejścia. Firmy obawiają się dopuszczenia osób niedoświadczonych do systemów produkcyjnych, gdzie błąd (np. w konfiguracji zapory sieciowej lub bazy danych) może spowodować ogromne straty finansowe dla danej firmy.
Ponadto, podana w projekcie podstawy programowej lista miejsc praktyk jest bardzo szeroka, ale mało specyficzna:

- Centra danych (Data Centers) ze względu na rygorystyczne normy (np. ISO 27001), wstęp do stref serwerowych dla osób postronnych (a tym bardziej praktykantów) jest niemal niemożliwy. Uczeń może najwyżej obserwować

pracę biurową, co nie realizuje efektów kształcenia takich jak utwardzanie systemów czy konfiguracja przełączników.
- Operatorzy infrastruktury krytycznej - to miejsca o najwyższym rygorze bezpieczeństwa. Realna szansa na odbycie tam praktyk przez 17-latka, który miałby aktywnie konfigurować systemy, jest bliska zeru.

Projekt podstawy programowej dopuszcza realizację do 50% wymiaru praktyk (140 z 280 godzin) w Branżowym Centrum Umiejętności. Jest to jedyny realistyczny zapis, który ratuje tę podstawę, o ile przynajmniej w każdym województwie.zostanie powołane BCU, które będzie realizowało warunki zbliżone do zadań zawodowych przewidzianych w zawodzie technika cyberbezpieczeństwa, mogą stworzyć środowiska symulacyjne (poligony cybernetyczne), gdzie uczeń bezpiecznie i legalnie przećwiczy ataki i obronę bez narażania realnych systemów.

5.2. Sugerowane nowe miejsca odbywania praktyk zawodowych

Biorąc pod uwagę zdiagnozowane wcześniej bariery (wiek uczniów, brak poświadczeń bezpieczeństwa, rygory ochrony danych), zapisy dotyczące miejsc odbywania praktyk zawodowych wymagają urealnienia i poszerzenia o podmioty, które dysponują odpowiednią infrastrukturą edukacyjną lub testową. Obecna lista jest zbyt ogólna i koncentruje się na podmiotach operacyjnych, które często mają „zamknięte drzwi” dla osób postronnych.

Należałoby rozważyć dodanie podmiotów, które z założenia zajmują się symulacją ataków i obrony, co rozwiązuje problem dostępu do systemów produkcyjnych. Są to:
- Centra Operacji Bezpieczeństwa (SOC) w formule MSSP (Managed Security Service Provider) - firmy świadczące usługi bezpieczeństwa dla wielu klientów często posiadają wyseparowane środowiska testowe,
- laboratoria testowe i badawcze (R&D) producentów rozwiązań cybersecurity - firmy tworzące antywirusy, firewalle czy systemy EDR, gdzie uczeń może brać udział w testowaniu nowych wersji oprogramowania,
- Akademickie Centra Cyberbezpieczeństwa - uczelnie wyższe posiadające laboratoria sieciowe, które mogą przyjmować praktykantów w ramach umów partnerskich ze szkołami.

W branży IT praca zdalna jest standardem. Podstawa powinna to uwzględniać, aby umożliwić współpracę z najlepszymi firmami z całego kraju. Dlatego też warto rozszerzyć zapisy do realizacji praktyk zawodowych w formule zdalnej (Virtual Internship) realizowane na wydzielonych, chmurowych środowiskach typu sandbox należących do pracodawcy. Świetnie się sprawdzi dopuszczenie uczniów do udziału w programach Bug Bounty - pod nadzorem mentora z firmy zewnętrznej uczeń mógłby uczyć się identyfikacji podatności w ramach kontrolowanych programów nagród za błędy. Jako miejsca praktyk zawodowych warto rozważyć Software

House'y z działami Quality Assurance (QA), ze szczególnym uwzględnieniem testowania bezpieczeństwa aplikacji (Security QA).

Propozycja zmian:
W podstawie programowej należy:

- dodać zapis, że praktyki zawodowe mogą odbywać się w formie symulowanej (np. na poligonach typu Cyber Range) w ramach porozumień z firmami, co rozwiąże problem braku poświadczeń bezpieczeństwa,
- zwiększyć limit realizowanych praktyk zawodowych w BCU do 100% w sytuacjach, gdy szkoła udokumentuje brak możliwości zapewnienia bezpiecznych miejsc praktyk u operatorów infrastruktury krytycznej,
- uwzględnić możliwości odbywania części praktyk zawodowych u pracodawcy w formule zdalnej na wydzielonych środowiskach testowych (sandbox), co jest standardem pracy w branży IT.

Sugerowany zapis miejsc realizacji praktyk zawodowych:
“Miejsce realizacji praktyk zawodowych: jednostki samorządu terytorialnego; przedsiębiorstwa usługowe IT; centra danych; centra operacji bezpieczeństwa (SOC); laboratoria badawczo-rozwojowe producentów oprogramowania zabezpieczającego; poligony cybernetyczne (Cyber Range); operatorzy infrastruktury krytycznej; podmioty prowadzące programy weryfikacji podatności (Bug Bounty); branżowe centra umiejętności; uczelnie wyższe prowadzące laboratoria cyberbezpieczeństwa. Dopuszcza się realizację praktyk w formie hybrydowej lub zdalnej na dedykowanych platformach laboratoryjnych zapewnionych przez pracodawcę.”

6. Liczba godzin w cyklu kształcenia

W zaproponowanym układzie godzin w projekcie podstawy programowej w zawodzie technik cyberbezpieczeństwa nie da się zrealizować kwalifikacji INF.12 do końca klasy czwartej, więc egzamin zawodowy musi odbyć się w połowie klasy piątej. Jest to bardzo niekorzystne z punktu widzenia możliwości realizacji specjalizacji w klasie piątej, co pozwoli na dużo lepsze przygotowanie uczniów na rynek pracy, w szczególności w tak dynamicznie rozwijającej się branży informatycznej w zakresie cyberbezpieczeństwa.

Rekomendacja
Wystarczy obniżyć 30 godzin z cyklu kształcenia, które wynikną z dostosowania efektów kształcenia i kryteriów weryfikacji do właściwego zadeklarowanego poziomu PRK, aby było możliwe zrealizowanie kształcenia wynikające z podstawy programowej do końca klasy czwartej.

7. Sztuczna inteligencja

Analiza projektu podstawy programowej w zawodzie technik cyberbezpieczeństwa w zakresie sztucznej inteligencji (AI) wykazuje, że autorzy dokumentu dostrzegli jej znaczenie, ale potraktowali ją niemal wyłącznie jako narzędzie pomocnicze, pomijając aspekty AI jako obiektu ataków oraz elementu infrastruktury, który należy zabezpieczać. Sztuczna inteligencja pojawia się w obu kwalifikacjach w bardzo podobnym kontekście - głównie jako wsparcie procesów wytwórczych i analitycznych. Podstawa programowa traktuje AI jako „magiczne narzędzie", zapominając o zagrożeniach wynikających z jego stosowania i specyfice zabezpieczania systemów opartych na AI.

Brakuje przede wszystkim efektów kształcenia dotyczących bezpiecznego korzystania z publicznych modeli LLM. Uczeń, generując kod za pomocą AI (co podstawa nakazuje), może nieświadomie wysyłać do zewnętrznych serwerów fragmenty kodu zawierające sekrety firmowe, klucze API czy dane wrażliwe. Brakuje kryterium w zakresie stosowania zasad ochrony poufności danych podczas interakcji z publicznymi modelami AI.

Projekt podstawy programowej **pomija również krytyczny aspekt bezpieczeństwa danych wprowadzanych do modeli AI - wyciek tajemnic przedsiębiorstwa, prompt injection** (wstrzykiwanie instrukcji sterujących modelem) **oraz data poisoning** (zatruwanie danych treningowych). Skoro uczeń ma weryfikować bezpieczeństwo aplikacji, powinien wiedzieć, jak zabezpieczyć aplikację korzystającą z AI przed przejęciem nad nią kontroli poprzez złośliwe zapytania.

Brakuje też efektów kształcenia i kryteriów weryfikacji dotyczącego bezpiecznego i etycznego korzystania z narzędzi LLM w pracy technika. Dotyczy to **krytycznej oceny wyników dostarczanych przez AI**: podstawa nakazuje generować kod za pomocą AI, ale zbyt słabo kładzie nacisk na fakt, że AI może generować kod z błędami logicznymi lub podatnościami (tzw. halucynacje). Uczeń musi posiadać kompetencję w zakresie identyfikowania błędów i poprawia ich w kodzie wygenerowanym przez sztuczną inteligencję. Dotyczy to również tematu Deepfake - w części dotyczącej uwierzytelniania biometrycznego brakuje powiązania z zagrożeniem obejścia biometrii twarzy/głosu za pomocą AI.

Dlatego niezbędnym jest przeanalizowanie całego projektu podstawy programowej w zawodzie technik cyberbezpieczeństwa i uzupełnienie efektów kształcenia oraz kryteriów weryfikacji w zakresie **zasad ochrony prywatności i danych wrażliwych podczas korzystania z modeli LLM (zapobieganie wyciekom danych), identyfikacji ataków specyficznych dla systemów AI oraz obowiązkową weryfikację merytoryczną i bezpieczeństwa kodu generowanego automatycznie. Bez tych zapisów szkoła może kształcić nawyk bezkrytycznego ufania narzędziom AI, co w cyberbezpieczeństwie jest błędem kardynalnym.**

8. Odpowiedzialność prawna i etyka zawodu

Analiza zapisów dotyczących etyki i odpowiedzialności prawnej w przedstawionej podstawie programowej wykazuje, że choć autorzy dostrzegli potrzebę kształtowania postaw etycznych, zapisy te są rozproszone i mało konkretne w zderzeniu z rygorami prawnymi sektora cybersecurity.
Mimo fragmentarycznych zapisów w jednostkach wprowadzających (INF.11.2 i INF.12.2) i w zakresie kompetencji personalnych, projekt podstawy programowej w zawodzie technik cyberbezpieczeństwa pomija kluczowe ramy prawne, które determinują codzienną pracę technika:

- Brak osadzenia w Kodeksie karnym - w podstawie programowej jest wspominana „odpowiedzialność”, ale nie nakłada obowiązku znajomości konkretnych artykułów (np. Art. 267–269 kk dotyczących hackingu, niszczenia danych czy zakłócania pracy systemów). Uczeń musi wiedzieć, gdzie kończy się „test”, a zaczyna przestępstwo zagrożone karą więzienia.
- Pominięcie aspektów RODO i ochrony prywatności - w zawodzie, który polega na dostępie do logów (często zawierających dane osobowe), brak bezpośredniego odwołania do RODO (GDPR) i obowiązków administratora danych jest rażącym błędem.
- Odpowiedzialność cywilna i zawodowa - brakuje zapisów o odpowiedzialności za błędy w sztuce (np. nieumyślne doprowadzenie do przestoju systemu podczas testów), co wiąże się z ubezpieczeniem OC zawodowym i zapisami w umowach o pracę/zlecenia.
- Standardy raportowania (Responsible Disclosure) - choć wspomniano o „odpowiedzialnym ujawnianiu podatności”, brakuje procedur postępowania w przypadku znalezienia luki w systemach osób trzecich (aspekt prawny vs. etyczny).

Wskazane braki są szczególnie istotne dla osób niepełnoletnich. Uczeń technikum musi mieć świadomość, że narzędzia, których uczy się w szkole (np. skanery podatności, intercepting proxy), są narzędziami "podwójnego zastosowania" (dual-use). Brak silnego nacisku na ramy prawne może prowadzić do nieświadomego wejścia na drogę przestępczą „z ciekawości”.

Podsumowując, podstawa programowa w niewystarczającym stopniu precyzuje ramy prawne działalności technika cyberbezpieczeństwa. Należy uzupełnić efekty kształcenia o znajomość konkretnych przepisów Kodeksu Karnego oraz RODO. Etyka nie może być traktowana jedynie jako postawa (uwzględniona jako kompetencja personalna, realizowana wyłącznie przez metody dydaktyczne stwarzające uczniom warunki do nabywania kompetencji personalnych i społecznych, na którą podstawa programowa nie przeznacza w realizacji żadnych godzin), lecz musi być powiązana z twardymi procedurami prawnymi, szczególnie w zakresie testów penetracyjnych i dostępu do danych wrażliwych.

Propozycja uzupełnienia:
Proponujemy dodanie efektu kształcenia wraz z kryteriami weryfikacji o odpowiedzialności cywilnej za błędy techniczne, co jest standardem w pracy eksperckiej (PRK 5). Do rozważenia uszczegółowienie kryteriów weryfikacji w zakresie:

- prawa karnego - np. “identyfikuje czyny zabronione w zakresie cyberprzestępczości zgodnie z Kodeksem Karnym (przestępstwa przeciwko ochronie informacji i mieniu)”
- prywatności - np. “stosuje zasady minimalizacji danych i ochrony prywatności (Privacy by Design) zgodnie z przepisami o ochronie danych osobowych (RODO)”
- etyki zawodowej - np. “analizuje kodeksy etyczne organizacji branżowych (np. (ISC)², ISACA) i stosuje je w sytuacjach konfliktowych”
- procedur - np. opracowuje procedurę bezpiecznego i zgodnego z prawem raportowania znalezionych podatności (Vulnerability Disclosure Policy)”

9. Zasady dostępności cyfrowej i standard WCAG

Analiza przedstawionego projektu podstawy programowej dla zawodu technik cyberbezpieczeństwa pod kątem dostępności cyfrowej (standardu WCAG) wykazuje krytyczny brak merytoryczny, który stawia ten zawód w gorszej pozycji niż zawody pokrewne, takie jak technik informatyk czy technik programista. Zawód technik cyberbezpieczeństwa został przyporządkowany do tej samej branży co technik informatyk i technik programista. W nowoczesnym kształceniu zawodowym standard WCAG jest traktowany jako fundament tworzenia i administrowania treściami cyfrowymi. Wykluczenie go z podstawy technika cyberbezpieczeństwa sugeruje, że bezpieczeństwo aplikacji jest odizolowane od jej dostępności, co jest błędem merytorycznym. Brak standardu WCAG w tej podstawie to nie tylko błąd "estetyczny", ale przede wszystkim funkcjonalny. Chociaż projekt podstawy wspomina o atakach typu DoS/DDoS (dostępność usługi), to całkowicie pomija dostępność dla osób ze szczególnymi potrzebami. Absolwent kształcący się w kwalifikacjach INF.11 i INF.12 będzie pracował m.in. w jednostkach samorządu terytorialnego i podmiotach publicznych. Zgodnie z ustawą o Krajowym Systemie Cyberbezpieczeństwa oraz ustawą o dostępności cyfrowej, systemy te muszą być dostępne. Brak znajomości WCAG uniemożliwi technikowi rzetelny audyt bezpieczeństwa strony publicznej, który powinien obejmować również poprawność strukturalną.
W obecnej strukturze podziału jednostek efektów kształcenia standard WCAG jest niezbędny w następujących miejscach:

- INF.12.3. Implementacja aplikacji webowych - uczeń uczy się technologii HTML, CSS i JavaScript. Bez WCAG uczeń może tworzyć kod, który jest "bezpieczny" technicznie, ale nieużyteczny (niedostępny), co w sektorze publicznym jest traktowane jako błąd krytyczny.

- INF.12.4. Weryfikowanie bezpieczeństwa aplikacji - audyt aplikacji powinien obejmować poprawność semantyczną kodu. Błędy w strukturze HTML (brak etykiet, nagłówków) ułatwiają ataki typu clickjacking lub manipulację interfejsem, co łączy się bezpośrednio z bezpieczeństwem.
- INF.11.8 / INF.12.6. Język angielski zawodowy - dokumentacja techniczna standardów bezpieczeństwa (np. od NIST czy ENISA) coraz częściej integruje wytyczne dotyczące dostępności jako element standardu jakości.

Warto dodać, że w projekcie podstawy programowej nie należy podawać wersji standardu WCAG ze względu na jego dużą zmienność. Można rozważyć odwołanie się do ustawy o dostępności cyfrowej, bo ten dokument w Polsce jest nadrzędnym nad standardem WCAG.

Propozycje uzupełnienia:

- W INF.12.3 - w zakresie stosowania wytycznych standardu WCAG podczas projektowania i implementacji interfejsu aplikacji webowej,
- W INF.12.4 - w zakresie przeprowadzenia audytów dostępności cyfrowej aplikacji pod kątem zgodności z obowiązującymi przepisami prawa.

10. Procedowanie włączenia zawodu technik cyberbezpieczeństwa i brak systemowej spójności w branży informatycznej

Procedowanie nowego zawodu technik cyberbezpieczeństwa w całkowitym oderwaniu od niezbędnych zmian w zawodach technik informatyk oraz technik programista jest błędem strategicznym, który doprowadzi do chaosu w kształceniu w branży INF:

- Pułapka silosowości - cyberbezpieczeństwo nie jest odizolowaną dyscypliną, lecz horyzontalną warstwą kompetencji, która musi przenikać całe IT. Wprowadzanie nowoczesnej podstawy dla "cyber-specjalisty", przy jednoczesnym pozostawieniu technika informatyka w realiach przestarzałych technologii sieciowych, tworzy system "dwóch prędkości". Absolwenci tych zawodów będą operować na skrajnie różnych standardach bezpieczeństwa, co w środowisku pracy jest niedopuszczalne.
- Kanibalizacja zamiast synergii - brak równoległej aktualizacji zawodów pokrewnych sprawia, że nowy zawód "kradnie" jednostki efektów kształcenia z programisty czy informatyka (często w formie kopiuj-wklej), nie oferując w zamian unikalnej wartości dodanej. Prowadzi to do dezorientacji pracodawców, którzy nie będą wiedzieli, czym w praktyce różni się "weryfikacja kodu" u programisty od tej u technika cyberbezpieczeństwa.
- Ignorowanie realnych potrzeb rynku (IoT, AI, Cloud) - podczas gdy Ministerstwo skupia się na jednym, medialnym haśle "cyberbezpieczeństwo", rynek pracy zgłasza krytyczne zapotrzebowanie na zawody takie jak technik internetu rzeczy (IoT). To właśnie miliardy niezabezpieczonych urządzeń IoT stanowią dziś główne źródło zagrożeń (botnety, wycieki danych), a w polskim

systemie edukacji obszar ten praktycznie nie istnieje. Podobnie rzecz ma się z administracją systemami chmurowymi czy integracją rozwiązań AI, które powinny być fundamentem wszystkich zawodów IT, a nie tylko dodatkiem do jednego.

- Konieczność redefinicji zadań zawodowych - zamiast mnożyć nowe byty, Ministerstwo powinno najpierw dokonać audytu i redefinicji zadań zawodowych w istniejących zawodach. Współczesny technik informatyk musi być "bezpieczny" z definicji. Tworzenie osobnego zawodu bez naprawy fundamentów kształcenia informatycznego to próba montowania pancernych drzwi (cyberbezpieczeństwo) w budynku o kruchych ścianach (przestarzałe podstawy informatyki i programowania).

Wniosek:
Projekt wymaga wstrzymania i szerszego spojrzenia, uwzględniającego Mapy Kompetencji Cyfrowych czy Strategią cyfryzacji Państwa, która w sposób spójny i zintegrowany zaktualizuje całą branżę informatyczną (INF), uwzględniając nowe zawody (IoT) oraz realną współpracę między nimi. Ponadto projekt wymaga **opracowania i procedowania równolegle zawodów branży INF**, aby móc również opracować wspólne kwalifikacje czy ścieżki rozwoju między tymi zawodami.

Podsumowując, cyberbezpieczeństwo nie istnieje w próżni, lecz jest integralną częścią całego ekosystemu IT.

W imieniu grupy ekspertów

Joanna Ksieniewicz