

Niniejszy dokument zawiera ogólne wymagania bezpieczeństwa dla aplikacji, która jest przedmiotem RFI. Wymagania podzielone są na 9 obszarów:

- I. Architektura
- II. Kontrola dostępu
- III. Identyfikacja i uwierzytelnianie
- IV. Kontrola integralności i ochrona przed złośliwym oprogramowaniem
- V. Kryptografia
- VI. Logowanie zdarzeń i kontrola rozliczalności
- VII. Dokumentacja i szkolenia
- VIII. Ciągłość działania
- IX. Warunki wsparcia
- X. Zgodność ze standardem korporacyjnym

Prosimy o podanie informacji, czy aplikacja spełnia wymagania zawarte w tabeli. Możliwe jest również wprowadzenie dodatkowego komentarza o sposobie realizacji wymagania.

I. Architektura

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
1.1.	Architektura systemu musi być co najmniej trójwarstwowa przy czym podstawowy podział na warstwy musi uwzględniać: warstwa danych (np. bazy danych, pliki), warstwa aplikacyjna (serwery aplikacyjne), warstwa prezentacji (serwery obsługujące interfejsy użytkownika, dostęp kliencki do systemu).		

II. Kontrola dostępu

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
2.1.	Mechanizmy kontroli dostępu zaimplementowane w aplikacji muszą egzekwować zasadę ograniczonego dostępu - „wszystko, co nie jest wyraźnie dozwolone, jest zabronione”. Tym samym, muszą skutecznie blokować dostęp użytkowników do wszystkich chronionych usług, funkcji, danych i obiektów (plików, katalogów, danych w bazie danych, URLi) z wyłączeniem tych, do których posiadają przyznane uprawnienia.		

2.2.	Jeżeli w ramach aplikacji funkcjonują użytkownicy o zróżnicowanych uprawnieniach, mechanizmy kontroli dostępu zaimplementowane w tej aplikacji muszą zapewniać możliwość stosowania zasady minimalnych uprawnień, czyli możliwość przydzielania użytkownikom wyłącznie tych uprawnień, które są im niezbędne do wykonywania niezbędnych działań.		
2.3.	Dostęp do części administracyjnej aplikacji i części zarządzania treścią merytoryczną aplikacji nie może być udostępniony w sieci Internet. Dostęp do tych części może być możliwy wyłącznie z sieci wewnętrznych.		
2.4.	Jeżeli w ramach pojedynczej aplikacji lub wielu instancji tej samej aplikacji funkcjonują użytkownicy przetwarzający odrębne zestawy danych, niepowiązane ze sobą (np. aplikacja jest wykorzystywana przez niezależne podmioty), zaimplementowane mechanizmy kontroli dostępu muszą skutecznie separować dostęp do zestawów danych poszczególnych użytkowników aplikacji, niezależnie od posiadanych uprawnień. To znaczy, że zaimplementowana separacja musi funkcjonować zarówno na poziomie uprawnień zwykłych użytkowników, jak również na poziomie uprawnień użytkowników uprzywilejowanych zarządzających uprawnieniami lub danymi.		
2.5.	Aplikacja musi chronić bezpośrednie odniesienia do obiektów tak, aby tylko uprawnione obiekty lub dane były dostępne dla użytkownika (np. ochrona przed manipulacją bezpośrednimi odniesieniami do obiektów).		
2.6.	Wszystkie ustalone reguły kontroli dostępu do usług, funkcji, danych i obiektów muszą być wymuszane po stronie serwera. Ponadto, mechanizmy kontroli dostępu muszą skutecznie blokować dostęp do reguł i atrybutów kontroli dostępu, tak aby nie była możliwa ich nieautoryzowana modyfikacja przez użytkowników aplikacji, chyba, że są oni do tego uprawnieni.		
2.7.	Aplikacja musi logować zdarzenia dotyczące kontroli dostępu.		
2.8.	Mechanizmy kontroli dostępu zaimplementowane w aplikacji muszą utrzymywać aktualny stan uprawnień użytkowników i w przypadku zmiany, ich egzekwowanie powinno być realizowane w trybie natychmiastowym.		
2.9.	Wszystkie informacje i pliki konfiguracyjne związane z bezpieczeństwem aplikacji muszą być przechowywane w miejscach chronionych przed nieautoryzowanym dostępem.		
2.10.	Listowanie katalogów musi być wyłączone.		

2.11.	Mechanizmy kontroli dostępu zaimplementowane w aplikacji muszą skutecznie wymuszać ustalone reguły przepływu pracy w aplikacji (przyjęta kolejność realizacji zadań) oraz nałożone limity (wielkość załącznika, ilość operacji, ilość transakcji, ilość zmian, itp.).		
2.12.	Aplikacja powinna wykorzystywać sprawdzone mechanizmy kontroli dostępu, które oferowane są przez system operacyjny, powiązane usługi i zastosowany framework ⁱⁱ .		
2.13.	Jeżeli to możliwe, aplikacja powinna używać mechanizmu "sandbox" lub izolować procesy.		
2.14.	Aplikacja lub framework musi generować silne, losowe tokeny anty-CSRF unikalne dla użytkownika, jako część wszystkich istotnych operacji lub przy dostępie do wrażliwych danych. Aplikacja musi weryfikować obecność tego tokenu przy przetwarzaniu wskazanych zapytań.		
2.15.	Zdalne obiekty IFRAME oraz wymiana zasobów pomiędzy domenami nie mogą pozwalać na zawieranie dowolnych treści zewnętrznych. Flash, Silverlight lub polityka wymiany zasobów pomiędzy domenami innej aplikacji internetowej musi być skonfigurowana tak by zapobiegać nieuwierzytelnionemu dostępowi zdalnemu.		

III. Identyfikacja i uwierzytelnianie

Lp.	Opis	SPEŁNIA / NIE SPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
3.1.	Wszyscy użytkownicy aplikacji lub procesy wykonywane w imieniu tych użytkowników muszą być jednoznacznie zidentyfikowani przed uzyskaniem dostępu.		
3.2.	Wszyscy użytkownicy aplikacji lub procesy wykonywane w imieniu tych użytkowników muszą przechodzić proces uwierzytelnienia.		
3.3.	Aplikacja powinna wymagać, aby wszystkie funkcje operujące na czynnikach uwierzytelniania danej tożsamości (np. rejestracja, aktualizacja profilu, przypomnienie loginu/hasła), które mogą przywrócić dostęp do konta są co najmniej tak odporne na ataki jak główny mechanizm uwierzytelniający.		
3.4.	Wszystkie strony oraz zasoby muszą wymagać uwierzytelnienia za wyjątkiem tych specjalnie przeznaczonych na dostęp publiczny.		
3.5.	Muszą istnieć mechanizmy pozwalające na zarządzanie informacjami uwierzytelniającymi, tj. nadawaniem, zmianą, blokowaniem, czasem życia i przechowywaniem konta.		
3.6.	W przypadku innego sposobu uwierzytelniania niż Active Directory muszą istnieć mechanizmy pozwalające na konfigurację złożoności haseł dla użytkowników, tj. budowy hasła (małe, wielkie litery,		

Lp.	Opis	SPEŁNIA / NIE SPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
	znaki specjalne, cyfry), długości hasła oraz czasu życia hasła.		
3.7.	Muszą istnieć mechanizmy zapewniające kontrolę sesji uwierzytelnionego użytkownika (np. zamykanie nieaktywnych sesji po określonym czasie).		
3.8.	Mechanizmy uwierzytelniania użytkowników, które kończą pracę niepowodzeniem muszą to robić w sposób bezpieczny, tzn. nie może nastąpić dopuszczenie użytkownika do chronionych zasobów oraz wyjątek, który miał miejsce musi być prawidłowo obsługany. Poprzez prawidłową obsługę przyjmuje się odnotowane zdarzenia w logach i wyświetlenie użytkownikowi dostosowanego komunikatu zawierającego identyfikator błędu, bez ujawniania szczegółów technicznych funkcjonowania samej aplikacji lub jej otoczenia infrastrukturalnego.		
3.9.	Aplikacja musi mieć możliwość uwierzytelniania użytkowników w oparciu o powszechnie używane i popularne metody uwierzytelniania.		
3.10.	W przypadku kont uprzywilejowanych (np. administratorów) powinno być zastosowane dwuskładnikowe uwierzytelnianie.		
3.11.	Dane uwierzytelniające nie mogą być zaszywane (ang. hardcoding) w kodzie aplikacji lub przekazywane w parametrach adresu URL.		
3.12.	W aplikacjach dostępnych w sieciach publicznych (np. sieci Internet), uwierzytelnianie w celu dostępu do zasobów związanych z czynnościami administracyjnymi powinno być możliwe jedynie z sieci wewnętrznej.		
3.13.	Aplikacja musi posiadać mechanizm ochrony przez atakami siłowymi (ang. brute-force) ⁱⁱⁱ na dane uwierzytelniające, blokujący kolejne próby uwierzytelnienia na zdefiniowany okres czasu.		
3.14.	Wszystkie pola służące do wprowadzania haseł nie mogą pokazywać haseł użytkowników w czasie ich wpisywania oraz muszą mieć wyłączoną funkcję automatycznego uzupełnienia.		
3.15.	Mechanizm autouzupełniania haseł oraz wrażliwych danych (np. PIN) musi być wyłączony.		
3.16.	Wszystkie decyzje dotyczące wyników uwierzytelniania muszą być logowane, zarówno te zakończone sukcesem jak i te zakończone porażką.		
3.17.	Odpowiedzi aplikacji powinny zawierać nagłówki HTTP zwiększające poziom bezpieczeństwa użytkowników aplikacji.		
3.18.	Aplikacja powinna wykrywać oraz blokować próby zautomatyzowanego logowania się na konta użytkowników z jednym hasłem (tzw. bruteforce horyzontalny)		
3.19.	Aplikacja musi zapobiegać enumeracji użytkowników przez		

Lp.	Opis	SPEŁNIA / NIE SPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
	logowanie, resetowanie hasła lub funkcjonalność przypomnienia loginu.		
3.20.	Aplikacja nie może używać domyślnych haseł dla frameworka aplikacji lub jakichkolwiek komponentów użytych przez aplikację.		
3.21.	Funkcja przypomnienia hasła nie może blokować lub w inny sposób wyłączać konta dopóki użytkownik nie zmieni hasła z powodzeniem. Należy jednak zabezpieczać się przed uporczywymi próbami przypomnienia hasła.		
3.22.	Aplikacja nie może zezwalać na równoczesne działanie wielu sesji tego samego użytkownika pochodzących z różnych maszyn.		

IV. Kontrola integralności i ochrona przed złośliwym oprogramowaniem

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
4.1.	Muszą istnieć mechanizmy lub proces potwierdzający autentyczność instalowanej aplikacji (sprawdzanie sum kontrolnych).		
4.2.	Muszą istnieć mechanizmy potwierdzające spójność krytycznych plików w systemie.		
4.3.	Muszą istnieć mechanizmy zapewniające kontrolę integralności sesji użytkownika.		
4.4.	Aplikacja nie może zawierać złośliwego kodu (w tym kodu, który mógłby zostać użyty do nawiązywania sesji z pominięciem mechanizmów uwierzytelniania – np. backdoor).		
4.5.	Muszą istnieć mechanizmy zapewniające kontrolę wprowadzanych danych (w przypadku wprowadzania ciągów znaków - ich format i składnię).		
4.6.	Muszą istnieć mechanizmy kontroli rodzaju, zawartości, zakresu, nazwy, formatu i wielkości plików w przypadku funkcjonalności związanych z importem plików (upload) oraz pobieraniem plików (download).		
4.7.	Aplikacja musi mieć zaimplementowane mechanizmy enkodowania/escapowania ^{iv} danych wyjściowych.		
4.8.	Wszystkie niezaufane dane wyjściowe trafiające do interpreterów SQL muszą używać parametryzowanych interfejsów (ang. prepared statements), przygotowanych instrukcji lub być właściwie escapowane.		
4.9.	Wszystkie niezaufane dane wyjściowe, których wynikiem jest XML muszą używać parametryzowanych interfejsów lub być właściwie escapowane.		

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
4.10.	Wszystkie niezauwane dane używane w zapytaniach LDAP muszą być właściwie escapowane.		
4.11.	Wszystkie niezauwane dane używane, jako parametry poleceń systemu operacyjnego muszą być właściwie escapowane.		
4.12.	Wszystkie niezauwane dane przekazywane do interpreterów innych niż ww., również muszą być właściwie escapowane.		
4.13.	Wszystkie połączenia wykorzystywane do komunikacji z innymi elementami aplikacji lub systemami podmiotów zewnętrznych, przez które przesyłane są dane wrażliwe, muszą być zabezpieczone kryptograficznie oraz strony takich połączeń muszą być uwierzytelnione.		
4.14.	Czas w aplikacji musi być synchronizowany z zaufanym źródłem czasu.		
4.15.	Naruszenia integralności muszą być logowane oraz objęte właściwą obsługą.		
4.16.	Aplikacja musi mieć możliwość współpracy z zastosowanymi wewnątrznie ustawieniami zabezpieczeń klienta WWW (używanej w Organizacji przeglądarki WWW).		
4.17.	Aplikacja nie może wysyłać żadnych informacji do serwerów w sieci Internet (w szczególności informacji o charakterze poufnym).		
4.18.	W aplikacji musi być możliwość pobierania uaktualnień z wewnętrznego serwera (zalecane, aby odbywało się to w sposób automatyczny).		
4.19.	Środowisko uruchomieniowe aplikacji musi wykorzystywać mechanizmy zapobiegające wystąpieniu błędów przepełnienia bufora pamięci.		
4.20.	W przypadku, gdy w aplikacji wykorzystywane są zewnętrzne biblioteki czy framework-i należy wykorzystywać wyłącznie aktualne wersje.		
4.21.	Aplikacja nie może polegać na skryptach znajdujących się na serwerach firm trzecich. Gdy jednak jest to konieczne, wykonawca musi o tym fakcie powiadomić ORLEN S.A. i uzyskać pisemną zgodę.		
4.22.	Dostawca powinien dostarczyć aplikację razem z kodem źródłowym.		
4.23.	Kod źródłowy aplikacji musi zostać poddany analizie bezpieczeństwa.		

V. Kryptografia

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
5.1.	Transmisja informacji wrażliwych przekazywanych za pośrednictwem publicznych sieci musi odbywać się wyłącznie w postaci zaszyfrowanej i tylko z wykorzystaniem protokołów komunikacji zapewniających wysoki poziom poufności.		
5.2.	Składowanie i przetwarzanie wrażliwych informacji w systemach, aplikacjach i bazach danych powinno być możliwe jedynie w postaci zaszyfrowanej (np. z użyciem natywnych mechanizmów szyfrujących).		
5.3.	Dane wrażliwe nie mogą być nigdzie przechowywane w postaci jawnej. System powinien zapewniać kryptograficzne środki ochrony przechowywanych danych wrażliwych.		
5.4.	Składowanie informacji uwierzytelniającej (hasła, tokeny oraz klucze prywatne) musi odbywać się jedynie w zabezpieczonej przestrzeni (np. zewnętrznym serwerze uwierzytelniającym, urządzeniu HSM ^v).		
5.5.	Przechowywanie informacji uwierzytelniającej w pamięci aplikacji musi odbywać się w sposób uniemożliwiający proste jej odczytanie (np. przechowywanie hasła w postaci jawnego tekstu jest niedopuszczalne).		
5.6.	Należy stosować mechanizmy zapobiegające przechowywaniu wrażliwych informacji po stronie użytkownika.		
5.7.	Aplikacja WWW udostępniona w sieci Internet (szczególnie służąca do komercyjnego udostępniania usług) musi stosować certyfikaty SSL wystawione przez zewnętrzne zaufane centrum certyfikacji. Dla szczególnie istotnych usług powinien to być certyfikat typu Extended Validation Certificate (EV SSL, ang. certyfikaty rozszerzonej walidacji).		
5.8.	Podjęcie zwiększona ilość zapytań informacyjnych lub zapytań wywołujących krytyczne transakcje muszą być blokowane.		
5.9.	Obsługa protokołu HTTP/HTTPS musi następować w zgodzie z najlepszymi praktykami bezpieczeństwa (np. ustawianie wymaganych nagłówków HTTP oraz ich atrybutów, konfiguracja protokołu TLS ^{vi}).		
5.10.	Stosowane moduły kryptograficzne muszą pracować w trybie zapewniającym należyty stopień bezpieczeństwa (stosowanie odpowiedniego źródła losowości, należyta obsługa błędów, zgodność z normami, standardami i wewnętrznymi politykami, a także najlepszymi praktykami w dziedzinie kryptografii):		

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
5.11.	Aplikacja powinna mieć możliwość implementacji polityk określających zarządzanie kluczami kryptograficznymi (np. generowanie, rozprowadzanie, unieważnianie, w jaki sposób wygasają).		
5.12.	Musi istnieć metoda usunięcia z aplikacji każdego typu wrażliwych danych na końcu ich wymaganego okresu ważności.		
5.13.	Jeżeli dane transmitowane są w tunelu SSL VPN, to musi być on zgodny ze specyfikacją transmisji danych protokołem HTTPS z użyciem obustronnego uwierzytelniania za pomocą certyfikatów PKI (niekwalifikowanych).		
5.14.	Jeżeli dane transmitowane są w tunelu IPSec VPN ^{vii} , to musi on spełniać następujące wymagania: <ul style="list-style-type: none"> Tryb pracy IPSec: ESP w trybie tunelowym; Protokół negocjacji parametrów: IKE; Metoda uwierzytelniania stron: <ul style="list-style-type: none"> Użytkownicy: <ul style="list-style-type: none"> Certyfikat PKI (niekwalifikowany); Systemy: <ul style="list-style-type: none"> Certyfikat PKI (niekwalifikowany); Symetryczne algorytmy szyfrowania: <ul style="list-style-type: none"> AES; Minimalna długość klucza: 128 bitów. Funkcje skrótu: HMAC-SHA-256 lub dłuższa; Grupy Diffie-Hellman: <ul style="list-style-type: none"> Modular Exponential (MODP), grupy 5, lub 14-18. Elliptic Curve Group over GF[2^N] (EC2N), grupa 4; Tryb negocjacji w fazie I: <ul style="list-style-type: none"> Main mode, Aggressive mode (zabroniony); Czas ważności kluczy: 3600 sekund. 		
5.15.	Rekomendowane jest składowanie danych na dyskach twardych w postaci zaszyfrowanej algorytmem zapewniającym należyty stopień poufności np. AES-128.		
5.16.	Parametry wykorzystywanych certyfikatów PKI powinny spełniać poniższe wytyczne: <ul style="list-style-type: none"> Algorytmy asymetryczne: RSA Długość kluczy: 2048 bitów Algorytm podpisu: SHA2-with-RSA Maksymalny okres ważności certyfikatów: 1 rok Regeneracja kluczy w trakcie odnowienia certyfikatów 		

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
5.17.	Parametry haseł użytkowników aplikacji webowych powinny spełniać poniższe wytyczne: <ul style="list-style-type: none"> • Hasła tworzone i przechowywane z użyciem soli • Długość soli: co najmniej 32 bajty • Sól^{viii} generowana losowo • Inna sól dla każdego konta • Dopuszczalne algorytmy dla zabezpieczania haseł: <ul style="list-style-type: none"> ○ Funkcje skrótu: SHA256, SHA512 ○ Scrypt, PBKDF2 (liczba iteracji: 64 000). 		
5.18.	W przypadku budowy aplikacji z wykorzystaniem usług serwisowych (web services) należy stosować mechanizmy zapewniające poufność i integralność przesyłanych danych.		

VI. Logowanie zdarzeń i kontrola rozliczalności

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
6.1.	Aplikacja musi rejestrować wszystkie zdarzenia, które mogą być pomocne w monitorowaniu poziomu bezpieczeństwa, analizie, ewentualnym dochodzeniu lub zgłoszeniu objawów naruszenia bezpieczeństwa, w tym jako min.: <ul style="list-style-type: none"> • udane i nieudane próby uwierzytelnienia, • założenie i usunięcie konta użytkownika • zablokowanie i odblokowanie konta użytkownika • aktywacja i dezaktywacja kont użytkowników, • zmiany ról i uprawnień użytkowników, • zmiany parametrów konfiguracyjnych, • próby przekraczania uprawnień. 		
6.2.	Zarejestrowane zdarzenia muszą mieć określoną strukturę, zakres oraz miejsce składowania.		
6.3.	Zarejestrowane zdarzenia muszą być przechowywane w bezpieczny sposób i chronione przed nieautoryzowanym ujawnieniem lub modyfikacją przez zdefiniowany okres czasu.		
6.4.	Wszyscy użytkownicy aplikacji muszą być jednoznacznie zidentyfikowani (poprzez przydzielanie unikatowych identyfikatorów).		
6.5.	Wszystkie operacje wykonywane w aplikacji muszą być jednoznacznie powiązane z odpowiedzialnymi za ich wykonanie użytkownikami.		

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
6.6.	Komunikaty błędów widoczne dla użytkowników nie mogą zawierać danych potencjalnie wrażliwych (np. dane osobowe, identyfikatory sesyjne użytkowników, dane nt. konfiguracji serwera, kod aplikacji, ślady stosu).		
6.7.	Logowane zdarzenia nie mogą zawierać danych wrażliwych (np. dane osobowe, identyfikatory sesyjne, dane konfiguracyjne serwera, hasła)		
6.8.	Moduł służący do przeglądania logów musi posiadać mechanizmy ochrony przez wykonaniem (potraktowaniem jako kod) niezauważanych danych będących treścią logów aplikacji.		
6.9.	W przypadku uwierzytelniania w oparciu o identyfikator i hasło, aplikacja musi wyświetlać ogólny komunikat, z którego nie będzie można wywnioskować czy błędnie został podany identyfikator czy hasło dostępowe.		
6.10.	Oprogramowanie musi używać jednej implementacji logowania zdarzeń aplikacyjnych.		
6.11.	Wszystkie niedrukowalne znaki oraz separatory pól muszą być poprawnie enkodowane w dziennikach zdarzeń.		
6.12.	Logowanie zdarzenia musi wystąpić przed wykonaniem akcji. Jeżeli logowanie się nie powiodło (np. przez brak miejsca na dysku, brakujące uprawnienia) aplikacja powinna kończyć pracę bezpiecznie.		

VII. Dokumentacja i szkolenia

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
7.1.	Dostarczenie dokumentacji projektu aplikacji.		
7.2.	Dostarczenie dokumentacji instalacji aplikacji.		
7.3.	Dostarczenie dokumentacji konfiguracji aplikacji i oprogramowania zawierającej opis zastosowanych zabezpieczeń, które wynikały z przedstawionych wymagań oraz schemat architektury rozwiązania (z wymienionymi portami i protokołami używanymi do komunikacji i transmisji informacji).		
7.4.	Dostarczenie dokumentacji użytkowej (eksploatacyjnej) aplikacji z podziałem dla odbiorców (użytkowników biznesowych i technicznych).		
7.5.	Dostarczenie dokumentacji związanej z usystematyzowaniem obsługi błędów i incydentów bezpieczeństwa.		

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
7.6.	Dostarczenie dokumentacji związanej z innymi elementami, np. procedurami pracy (monitorowanie poprawności działania, tworzenia kopii zapasowych i odtwarzania ich po awarii), zgodności z regulacjami (np. Ustawą o Ochronie Danych Osobowych), zastosowanymi interfejsami do innych aplikacji i systemów.		
7.7.	Dostarczenie szkolenia lub szkoleń związanych z zapewnianiem bezpieczeństwa aplikacji (integralności, poufności i dostępności).		

VIII. Ciągłość działania

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
8.1.	Każdy element dostarczanej aplikacji (np. front-end, repozytorium danych) powinien umożliwiać zastosowanie dodatkowych elementów realizujących redundancje.		
8.2.	Dla wszystkich elementów krytycznych dla zapewnienia ciągłości działania aplikacji powinna zostać zastosowana nadmiarowość (w postaci zapewnienia redundancji tych zasobów).		
8.3.	W przypadku aplikacji intranetowych aplikacja musi być zgodna z przeglądarką z aktualnego Korporacyjnego Standardu Informatycznego(KSI). Zgodność aplikacji internetowej z wiodącymi na rynku przeglądarkami internetowymi (Internet Explorer w wersji 7,8,9,10,11 oraz najaktualniejszymi wersjami: Google Chrome, Mozilla Firefox, Safari, Opera) powinna być zapewniona przez ustalony okres czasu.		
8.4.	Zgodność aplikacji z wiodącymi na rynku systemami operacyjnymi powinna być zapewniona przez ustalony okres czasu.		
8.5.	Wszystkie podatności krytyczne, wysokie i średnie powinny być wyeliminowane przed fazą produkcyjną.		
8.6.	Musi być zapewniona możliwość logowania się administratora do systemu operacyjnego na którym uruchomiona jest aplikacja lub jej fragment.		

IX. Warunki wsparcia

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
9.1	Firma dostarczająca aplikację powinna zapewnić wsparcie.		
9.2	W umowie serwisowej uwzględnione powinny być elementy takie jak czas reakcji, czas naprawy, RPO, RTO		
9.3	Zgłoszenia powinny być przyjmowane przynajmniej jedną z dróg: telefoniczną, za pomocą e-mail czy dedykowanego serwisu internetowego.		
9.4	Powinna zostać ustalona priorytetyzacja zgłoszeń oraz czas reakcji dotyczący zgłoszeń o danym priorytecie.		
9.5	Wszystkie podatności muszą być naprawione przez dostawcę. Podatność rozumiana jest jako opisana i wciągnięta do ewidencji luka w zabezpieczeniach opublikowana w powszechnie znanych, publicznych i uznawanych za wiarygodne katalogach takich, jak CVE, CCE, biuletyn bezpieczeństwa Microsoft i innych (np. dostępnych poprzez http://web.nvd.nist.gov/view/vuln/search).		

X. Zgodność ze standardem korporacyjnym

Lp.	Opis	SPEŁNIA / NIESPEŁNIA	Uwagi/Uzasadnienie warunkowego odstąpienia
10.1	Aplikacja powinna być pisana w sposób kompatybilny z aktualnie obowiązującym Korporacyjnym Standardem Informatycznym w ORLEN S.A.		

Przypisy

ⁱ **URL (ang. Uniform Resource Locator)** – oznacza ujednolicony format adresowania zasobów (informacji, danych, usług) stosowany w Internecie i w sieciach lokalnych.

ⁱⁱ **Framework (platforma programistyczna)** - jest szkieletem do budowy aplikacji. Definiuje on strukturę aplikacji oraz ogólny mechanizm jej działania, a także dostarcza zestaw komponentów i bibliotek ogólnego przeznaczenia do wykonywania określonych zadań. Programista tworzy aplikację, rozbudowując i dostosowując poszczególne komponenty do wymagań realizowanego projektu, tworząc w ten sposób gotową aplikację.

-
- iii **Atak siłowy (ang. brute-force)** - określenie algorytmu, który opiera się na sukcesywnym sprawdzeniu wszystkich możliwych kombinacji hasła dostępowego.
- iv **Enkodowanie/Escapowanie** – mechanizm polegający na usuwaniu i/lub kodowaniu znaków specjalnych w sposób który uniemożliwi systemowi ich interpretację w sposób inny niż przewidział to programista.
- v **HSM (ang. Hardware Security Module)** – Sprzętowy moduł kryptograficzny. System służący między innymi do bezpiecznego przechowywania kluczy poufnych.
- vi **TLS (ang. Transport Layer Security)** – przyjęte jako standard w Internecie rozwinięcie protokołu **SSL (ang. Secure Socket Layer)**, zaprojektowanego pierwotnie przez Netscape Communications. TLS zapewnia poufność i integralność transmisji danych, a także uwierzytelnienie serwera, a niekiedy również klienta. Opiera się na szyfrowaniu asymetrycznym oraz certyfikatach X.509.
- vii **VPN/IPsec (Virtual Private Network)** - wirtualna sieć prywatna. Można ją opisać jako "tunel", przez który płynie ruch w ramach sieci prywatnej pomiędzy klientami końcowymi za pośrednictwem publicznej sieci (takiej, jak Internet) w taki sposób, że węzły tej sieci są przezroczyste dla przesyłanych w ten sposób pakietów. Taki kanał może opcjonalnie kompresować lub szyfrować w celu zapewnienia lepszej jakości lub większego poziomu bezpieczeństwa przesyłanych danych.
- viii **Sól (ang. Salt)** – są to dane dodawane podczas szyfrowania lub przygotowywania skrótu wiadomości (w tym skrótu hasła).