Specyfikacja Oprogramowania

Przedmiotem zamówienia jest dostawa licencji czasowych w modelu subskrypcyjnym oprogramowania w rozwiązaniu SaaS, służącego do monitorowania infrastruktury aplikacyjnej Zamawiającego, składającego się na oprogramowanie klasy APM (ang. Application Performance Monitoring) w na okres 36 miesięcy wraz z Usługą Wsparcia Oprogramowania na okres trwania subskrypcji, spełniający poniższe funkcjonalności:

Oprogramowanie musi zapewniać monitorowanie zachowania i wydajności aplikacji, baz danych oraz monitorowanie doświadczeń użytkownika końcowego wraz z pełną diagnostyką działania na stacji roboczej w zakresie działania monitorowanych aplikacji i usług, oraz weryfikacji podatności bibliotek programistycznych stosowanych w środowisku aplikacyjnym. Oprogramowanie musi monitorować i weryfikować wszystkie serwery aplikacyjne, bazodanowe wchodzące w skład środowiska, również dla części maszyn wirtualnych.

1. **Wymagania funkcjonalne Oprogramowania do monitorowania środowiska produkcyjnego i uat :**
   1.1. Monitorowanie aplikacji ma odbywać się w sposób ciągły, z wykorzystaniem oprogramowania monitorującego. Część centralna oprogramowania może być udostępniona w modelu SaaS.
   1.2. Rozwiązanie oparte o model SaaS musi zapewniać poziom bezpieczeństwa potwierdzony min. Certyfikatem SOC-2 lub równoważnym. Rozwiązanie powinno być zgodne z zasadami dyrektywy GDPR oraz zapewniać, że dane w SaaS będą przechowywane oraz przetwarzane na terenie Unii Europejskiej. Komunikacja pomiędzy komponentami rozwiązania w SaaS a komponentami w infrastrukturze Zamawiającego musi być szyfrowana z wykorzystaniem min. AES-256 lub równoważnym, komunikacja może się odbywać jedynie jednokierunkowo – od strony komponentów zlokalizowanych w infrastrukturze Zamawiającego do SaaS.
   1.3. Oprogramowanie musi być dostępne w postaci interfejsu graficznego z poziomu przeglądarki internetowej.
   1.4. Dostęp do Oprogramowania musi być zabezpieczony hasłem. Autentykacja i autoryzacja w Oprogramowaniu ma umożliwiać kontrolę dostępu opartą na rolach (RBAC).
   1.5. Oprogramowanie wykorzystywane dla świadczenia usługi musi umożliwiać integrację bazy użytkowników z LDAP.
   1.6. Dostęp do Oprogramowania musi zapewniać zabezpieczenie dostępu z poziomu operatora i użytkownika Oprogramowania za pomocą protokołu HTTPS.
   1.7. Elementy oferowanego rozwiązania muszą w zakresie komunikacji (wewnętrznej i zewnętrznej) umożliwiać wykorzystywanie protokołów bezpieczeństwa, przynajmniej SSL.
   1.8. Oprogramowanie musi posiadać możliwość integracji z wykorzystywanym przez Zamawiającego systemem wielostopniowej autentykacji (MFA) w zakresie logowania do części centralnej systemu.
2. **Funkcje Oprogramowania:**
   2.1. **Musi zapewniać** możliwość uruchomienia monitoringu dla aplikacji pracujących przynajmniej na następujących systemach operacyjnych:
      a. AIX
      b. Linux:
         - CentOS
         - Debian
         - Fedora
         - openSUSE Leap
         - Red Hat Enterprise Linux
         - SUSE Linux Enterprise
         - Ubuntu
      c. Windows:
         - 2012
         - 2012 R2
         - 2016
         - 2019
         - 2022
   2.2. Oprogramowanie musi zapewniać możliwość monitorowania wielowarstwowych aplikacji wykonanych w następujących technologiach:
      a. Java
      b. .Net
      c. PHP

d. Node.js
e. C/C++
f. Python
g. Go

2.3. Oprogramowanie, na podstawie wykrytych przepływów, w sposób automatyczny powinno pozwolić na odwzorowanie w formie graficznej monitorowanego systemu, obrazując powiązania i zależności monitorowanych komponentów i procesów oraz ich wzajemną komunikację, w szczególności uwzględniając takie warstwy jak serwery aplikacyjne, bazy danych, zewnętrzne serwisy i kolejki. W przypadku wykrycia odstępstwa od normy skutkującej wygenerowaniem alertu monitorowany komponent, musi zostać oznaczony na wizualizacji w sposób jednoznacznie wskazujący na wystąpienie problemu w danym miejscu.

2.4. Oprogramowanie musi wykrywać i monitorować przebieg wszystkich transakcji przepływających przez aplikację w sposób automatyczny oraz oferować możliwość ręcznego dostosowania sposobów wykrywania i monitoringu transakcji.

2.5. Oprogramowanie musi wspierać definiowanie własnych transakcji biznesowych na podstawie spersonalizowanych reguł dopasowania, opartych o:
a. URL
b. wartość parametru z nagłówka HTTP,
c. wartość parametru z zapytania GET lub POST,
d. wykonanie konkretnej metody w kodzie Java lub .NET,
e. wywołanie konkretnej usługi WebService.

2.6. Oprogramowanie musi wspierać automatyczne wykrywanie rodzajów komunikacji pomiędzy wykrytymi komponentami monitorowanych aplikacji, w tym wspieranie śledzenia transakcji wykorzystujących co najmniej następujące technologie synchroniczne i asynchroniczne:
a. HTTP
b. REST
c. SOAP/XML
d. JMS

2.7. Oprogramowanie musi oferować możliwości uzyskania następujących informacji o wybranych transakcjach:
a. drzewo wywołania kodu oprogramowania w ramach transakcji uwzględniając nazwy wywoływanych metod, zarówno dla wątków wywoływanych synchronicznie, jak i asynchronicznie wraz z czasem wykonania pojedynczych metod,
b. czasy odpowiedzi serwera do aplikacji klienckiej, jak i całkowity czas wykonania transakcji po stronie serwera (wątków synchronicznych oraz asynchronicznych),
c. zapytania SQL wykonane w ramach transakcji z możliwością uzyskania informacji o użytych w nich zmiennych lub celowego ich maskowania,
d. wartości parametrów wywołania wskazanych metod,
e. wartości zwracane przez wskazane metody.

2.8. Oprogramowanie musi umożliwiać korelację transakcji realizowanych przez monitorowane komponenty z odpowiadającymi im danymi infrastrukturalnymi, bazodanowymi i sesją użytkownika końcowego.

2.9. Oprogramowanie musi udostępniać reguły powiadamiania w przypadku wykrycia problemów z wydajnością w aplikacji lub innych anomalii w oparciu o automatycznie wygenerowane linie bazowe, lub statyczne wartości.

2.10. Oferowane Oprogramowanie musi automatycznie, na podstawie danych bazowych/wzorcowych wykrywać problemy związane co najmniej z:
a. wydłużeniem czasów odpowiedzi poszczególnych usług po stronie serwerowej,
b. zwiększeniem poziomu problemów dla poszczególnych usług po stronie serwerowej,
c. wydłużeniem czasów odpowiedzi dla poszczególnych akcji wykonywanych przez użytkownika końcowego na aplikacji WWW lub aplikacji mobilnej,
d. zwiększeniem poziomu problemów dla poszczególnych akcji wykonywanych przez użytkownika końcowego na aplikacji WWW lub aplikacji mobilnej,
e. przeciążeniem CPU,
f. nadmiernym wykorzystaniem pamięci,
g. spadkiem wydajności dysków,
h. brakiem dostępności aplikacji.

2.11. Umożliwia definiowanie, konfigurację i modyfikację reguł, na podstawie których oprogramowanie generuje alerty. Oprogramowanie musi mieć możliwość wygenerowania alertu na podstawie zadanego odchylenia danej metryki na podstawie statycznego progu.

2.12. Na podstawie wygenerowanego alertu, Oprogramowanie musi umożliwiać wykonanie automatycznie następujących akcji:

a. Wysłanie powiadomienia do konkretnych użytkowników za pomocą wiadomości SMS lub e-mail,
b. Wysłać zapytanie HTTP o dowolnej treści na dowolny URL.

2.13. Oprogramowanie musi posiadać mechanizm przeciwdziałania generowania fałszywych alertów.
2.14. Pozwala zbierać i monitorować najbardziej wpływające na wydajność monitorowanej aplikacji zapytania SQL wykonywane z poziomu monitorowanej aplikacji z możliwością ich powiązania z transakcjami, które dane zapytania wykonują.
2.15. Ogranicza swój wpływ na monitorowane platformy i aplikacje m.in. poprzez inteligentne zbieranie informacji celem uniknięcia zbędnego zużywania zasobów.
2.16. Ma możliwość prezentowania na wykresach dowolnych metryk gromadzonych przez oprogramowanie.
2.17. Oprogramowanie musi pozwalać na tworzenie dowolnych niestandardowych pulpitów prezentujących gromadzone w ramach usługi dane, z poziomu interfejsu graficznego. Oprogramowanie musi umożliwiać nadawanie użytkownikom uprawnień wyświetlania lub edycji poszczególnych pulpitów.
2.18. Wykonawca w ramach dostarczonego oprogramowania musi w przypadku wykrycia problemu automatycznie wskazać możliwe przyczyny wystąpienia problemu.
2.19. Wykonawca musi gwarantować odpowiedni poziom dostępu do danych definiowany na poziomie nadawania uprawnień do dostarczonego Oprogramowania oparty o system ról i grup użytkowników (ang. Role-Based Access Control). Mechanizm konfiguracji uprawnień musi być dostępny w interfejsu graficznego, jak i z poziomu interfejsu API dostarczonego oprogramowania, służącego wykonywaniu usługi.
2.20. Oprogramowanie musi umożliwiać porównywanie działania aplikacji w różnych przedziałach czasowych na poziomie czasów odpowiedzi, liczby błędów, poziomu ruchu i tym podobnych.
2.21. Oprogramowanie musi zbierać informacje o wszystkich błędach i wyjątkach. Musi istnieć możliwość zobaczenia szczegółowych informacji na temat transakcji, w których wystąpił błąd bądź został wygenerowany wyjątek.
2.22. Oprogramowanie, poza domyślnym mechanizmem detekcji problemów, musi oferować możliwość konfiguracji tzw. wyjątków – odstępstw od reguły, pozwalające na odrzucenie błędów technicznych, które nie mają wpływu na biznesowe działanie aplikacji.
2.23. Dostarczone oprogramowanie musi zapewniać wyszukiwanie w zgromadzonym przez niego zbiorze danych dotyczących transakcji na podstawie definiowalnych filtrów, lub zapytań. Musi istnieć możliwość definiowania wielu filtrów lub zapytań wraz z możliwością ich zapisu celem ciągłego lub wielokrotnego użycia.
2.24. Dostarczone oprogramowanie musi pozwalać na użycie operatorów logicznych, wzorców, wyrażeń regularnych (REGEX) w filtrach lub zapytaniach używanych do przeszukiwania danych dotyczących transakcji.
2.25. Dostarczone oprogramowanie musi mieć możliwość tworzenia lub konfigurowania definiowanych przez Administratora, lub użytkownika aplikacji dodatkowych niestandardowych wtyczek monitorujących.
2.26. Oprogramowanie musi posiadać funkcjonalność logowania wszystkich aktywności użytkowników związanych ze zmianami konfiguracji. Logowanie musi umożliwiać jednoznaczne wskazanie osoby, która wykonała zmianę.
2.27. Dostarczone oprogramowanie - musi oferować także udokumentowany interfejs programistyczny (API) służący do konfiguracji Oprogramowania, pobierania danych, a w tym metryk historycznych.
2.28. Pozwala analizować wpływ zmian wersji oprogramowania na wydajność procesów, transakcji oraz wartość metryk związanych z obsługą użytkowników aplikacji, celem wskazania czy wprowadzane zmiany prowadzą do pożądanego stanu funkcjonowania aplikacji. Oprogramowanie musi oferować możliwość rejestracji zdarzenia wgrania nowej wersji aplikacji.

3. W zakresie monitorowania użytkownika końcowego, oprogramowanie musi spełniać poniższe wymagania techniczne i posiadać niżej wymienione funkcje:

3.1. Pozwala na monitorowanie sposobu działania aplikacji z perspektywy przeglądarek internetowych użytkowników końcowych w zakresie czasu odpowiedzi aplikacji i występujących błędów.
3.2. Przedstawia informacje w jaki sposób użytkownicy końcowi wchodzą w interakcję z aplikacją i w jaki sposób w niej nawigują.
3.3. Przedstawia wpływ sieci i czasu wczytywania aplikacji po stronie przeglądarki internetowej na doświadczenia użytkownika końcowego.

3.4. Pozwala na zbieranie danych dotyczących używanej przeglądarki, systemu operacyjnego, wykorzystywanego urządzenia i innych parametrów pozwalających na identyfikację jak aplikacja działa dla różnych grup użytkowników.
3.5. Musi umożliwiać powiązanie monitorowanej sesji użytkownika końcowego z interakcją z systemem i transakcjami realizowanymi przez system na poziomie sekwencji wywołanych metod i skorelowanych informacji infrastrukturalnych, od rozpoczęcia aktywności (np. dostęp do strony WWW), aż do jej zakończenia (np. odpowiedź bazy danych).
3.6. Musi umożliwiać automatyczne sprawdzanie dostępności i wydajności aplikacji poprzez cykliczne lub jednorazowe wykonywanie skryptu symulującego pracę użytkownika z możliwością monitorowania pracy użytkownika końcowego zarówno z wewnątrz, jak i z zewnątrz infrastruktury Zamawiającego.
3.7. Musi pozwalać na geolokalizację zdarzeń na bazie adresów IP oraz umożliwiać jej wizualizacje na mapie.
3.8. Musi pozwalać na mapowanie danych geolokalizacyjnych dla wewnętrznej adresacji IP poprzez wprowadzenie wymaganych danych przez interfejs web, plik z danymi geolokacyjnymi lub modyfikację agenta monitorującego użytkownika końcowego.
3.9. Pozwala na monitorowanie działanie aplikacji w wersji mobilnej dla systemów Android i iOS.
3.10. Umożliwia robienie zrzutów ekranów aplikacji mobilnych w systemach Android i iOS.
3.11. Musi pozwalać na integrację z narzędziami do monitorowania ścieżki sieciowej przebytej z różnych miejsc w sieci do monitorowanej aplikacji przeglądarkowej. Integracja ta musi pozwalać na wyświetlanie w dostarczanym narzędziu monitorującym, metryk dotyczących utraty pakietów, opóźnienia (ang. latency) oraz jitter na ścieżce pomiędzy wybranym miejscem w sieci a monitorowaną aplikacją przeglądarkową. Dodatkowo oprogramowanie musi posiadać możliwość automatycznego zmapowania monitoringu skonfigurowanym w narzędziu do monitorowania ścieżki sieciowej z aplikacjami monitorowanymi w dostarczanym narzędziu monitorującym na podstawie adresu URL, z którego dostępna jest monitorowana aplikacja.

4. W zakresie gromadzenia, monitorowania i analizy logów, oprogramowanie musi spełniać poniższe wymagania i posiadać niżej wymienione funkcje:
4.1. ostęp do logów aplikacyjnych i systemowych, przeszukiwanie ich i przeglądanie bez konieczności logowania na serwer monitorowany.
4.2. zbieranie danych wyodrębnianych z plików logów umieszczonych w wyszczególnionym katalogu.
4.3. rozpoznanie formatów czasu i daty w przetwarzanych danych z logów i normalizowanie ich do jednego wspólnego formatu.
4.4. zapewniać użytkownikowi posiadającemu uprawnienia samodzielne konfigurowanie reguł odczytu logów w celu umożliwienia analizy zebranych w nich informacji.

5. Wykonawca, w ramach świadczenia usługi monitorowania podatności bibliotek programistycznych zapewni:
5.1. Ciągłe monitorowanie istniejących, znanych podatności, istniejących lub takich, które pojawią się w czasie świadczenia usługi i zostaną sklasyfikowane za pomocą sygnatur podatności CVE oraz CVSS.
5.2. Zamawiający uzyska ciągły (24/7h) dostęp do bazy informacji na temat podatności w środowiskach monitorowanych przez System w taki sposób, aby mógł zaimplementować w procesy CI/CD uzyskane informację o:
   a) Stosowanych w Systemie bibliotekach programistycznych.
   b) Powiązaniu bibliotek w konkretnymi procesami aplikacyjnymi i wykonywanymi metodami.
   c) Podatnościach tychże bibliotek lub ich braku.
   d) Umiejscowieniu podatności w ramach aplikacji oraz infrastruktury serwerowej.
   e) Poziomie krytyczności wykrytych podatności ze wskazaniem, które z podatności mają charakter krytyczny, wysoki, średni oraz niski.
5.3. Wykonawca zapewni - ciągłą aktualizację informacji określonych w pkt 5.2.
5.4. Dostarczone oprogramowanie musi być oparte o dane min. jednej organizacji Threat Intelligence, dostępne w sposób ciągły bez konieczności instalowania żadnych komponentów sprzętowych w infrastrukturze Zamawiającego.

Załącznik nr 1 do Specyfikacji Oprogramowania

Wykaz infrastruktury Zamawiającego, podlegającej monitorowaniu w zakresie dostarczonego Oprogramowania.

1. Na infrastrukturę obsługującą produkcyjny system składają się następujące serwery:

System operacyjny	Rodzaj serwera	vCPU	RAM (w GB)	Stosowane Technologie
ePUE				
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
Red Hat Core OS	Aplikacyjny	16	128	OpenShift, Java
RHEL 7.9	Bazodanowy	24	64	PostgreSQL
RHEL 7.9	Bazodanowy	24	64	PostgreSQL
RedHat SSO				
RHEL 7.4	Aplikacyjny	16	24	Jboss, Java
RHEL 7.4	Aplikacyjny	16	24	Jboss, Java
RHEL 7.4	Aplikacyjny	16	24	Jboss, Java
RHEL 7.4	Aplikacyjny	16	24	Jboss, Java
RHEL 7.4	Bazodanowy	24	48	PostgreSQL
LIDER				
RHEL 6.7	Aplikacyjny	24	64	Jboss, Java
RHEL 8.9	Bazodanowy	12	64	Oracle DB
RED				
RHEL 7.6	Aplikacyjny	2	8	Java, NodeJS
RHEL 7.4	Bazodanowy	24	64	PostgreSQL
KeyCloak SSO*				
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java,
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java,
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java
RHEL 7.7	Bazodanowy	4	32	PostgreSQL
Web server				
RHEL 7.4	Infrastrukturalny	88	756	Nginx
RHEL 7.4	Infrastrukturalny	88	756	Nginx
Serwer Kolejek				
RHEL 8.6	Kolejkowy	4	16	Jboss AMQ
RHEL 7.9	Kolejkowy	4	24	Jboss AMQ

*) KeyCloak SSO zarówno na środowisku produkcyjnym, jak i UAT wykorzystuje wspólne serwery (wspólne środowisko OKD) w warstwie aplikacyjnej.

2. Na infrastrukturę obsługującą system UAT składają się następujące serwery:

System operacyjny	Rodzaj serwera	vCPU	RAM (W gb)	Stosowane Technologie
ePUE				
Red Hat Core OS	Aplikacyjny	25	192	OpenShift, Java
Red Hat Core OS	Aplikacyjny	25	192	OpenShift, Java
Red Hat Core OS	Aplikacyjny	25	192	OpenShift, Java
RHEL 7.9	Bazodanowy	12	40 GB	PostgreSQL
RedHat SSO				
RHEL 8.5	Aplikacyjny	16	32	Jboss, Java
RHEL 7.7	Bazodanowy	48	512	PostgreSQL
LIDER				
RHEL7.3	Aplikacyjny	8	16	Jboss, Java
RHEL 6.6	Bazodanowy	16	96	Oracle DB
RED				
RHEL 7.7	Aplikacyjny	2	8	Java, NodeJS
RHEL 6.5	Bazodanowy	10	64	PostgreSQL
KeyCloak SSO*				
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java
CentOS 7.8	Aplikacyjny	88*	756*	OKD,Jboss/Wildfly, Java
RHEL 7.3	Bazodanowy	4	8	PostgreSQL
Web server				
RHEL 7.4	Infrastrukturalny	4	32	Nginx
Serwer Kolejek				
RHEL 7.9	Kolejkowy	1	12	Jboss AMQ
RHEL 8.6	Kolejkowy	4	16	Jboss AMQ

*) KeyCloak SSO zarówno na środowisku produkcyjnym, jak i UAT wykorzystuje wspólne serwery (wspólne środowisko OKD) w warstwie aplikacyjnej.