Szczegółowy opis przedmiotu zamówienia

Oprogramowanie antywirusowe musi spełniać co najmniej następujące wymagania:

1. Agent musi obsługiwać funkcjonalności Next Generation EPP (Endpoint Protection Platform) oraz EDR (Endpoint Detection and Response) w jednym autonomicznym agencie, który do realizacji swoich funkcjonalności nie potrzebuje łączności z chmurą lub konsolą zarządzającą. Wymagane jest wsparcie dla systemów operacyjnych Windows, macOS i Linux.
2. Rozwiązanie musi być w stanie identyfikować zaawansowane zagrożenia, takie jak ataki bez plikowe, 0-day malware czy wykorzystywanie podatności posiadanego software/hardware bez korzystania z silników reputacji lub silników detekcji opartej o sygnatury. Przez silnik reputacyjny rozumiemy identyfikację zagrożeń z wykorzystaniem następujących elementów reputacji: adresy IP, DNS, URL, skróty/hashe. Rozwiązanie musi wykorzystywać statyczne oraz dynamiczne AI do identyfikacji zagrożeń, również tych które nie są wcześniej znane.
3. Agent musi być w pełni autonomiczny, co oznacza że jego działanie i funkcjonalność nie może być zależna od serwera zarządzania, chmury ani ŻADNYCH zasobów zewnętrznych od agenta. Wykrywanie i reagowanie na zaawansowane zagrożenia (0-day, bezplikowe, oparte na pamięci RAM , Exploity 0-Day, ransomware, cryptominers, lateral movement, APT) musi być możliwe w czasie rzeczywistym , nie może zależeć od stanu sieciowego stacji (agent musi realizować te same funkcjonalności w trybie online i offline) oraz nie może wymagać innego rodzaju zewnętrznych zasobów.
4. Moduły EPP / EDR oferowane w rozwiązaniu muszą automatycznie reagować na pojawiające się zagrożenia, łagodząc zagrożenia w czasie zbliżonym do rzeczywistego, w autonomiczny sposób, z następującymi opcjami odpowiedzi na zagrożenie, definiowane przez politykę bezpieczeństwa:
    1) Ostrzeżenie: taka notyfikacja musi być stała, nawet jeśli polityka nie jest w trybie ochrony;
    2) Zabij proces: Zatrzymuje procesy. Aktywna zawartość w dokumentach, plikach wykonywalnych i procesach podrzędnych jest zatrzymywana. Agent włącza funkcję zabicia procesu dla procesów, które działają wbrew normalnemu zachowaniu stacji końcowej lub nie pasują do działań aplikacji, w której ukrywa się proces.
    3) Kwarantanna: zatrzymuje procesy, szyfruje plik wykonywalny i przenosi go na ograniczoną ścieżkę. Jeśli zagrożenie jest znane, agent automatycznie je unieszkodliwia, zanim będzie można je wykonać.
    4) Odłącz się od sieci: (lub kwarantanna sieciowa lub izolacja sieciowa) Agent musi komunikować się tylko z konsolą zarządzającą. Stacja końcowa nie może komunikować się z innymi elementami w sieci. Wszystkie działania na konsoli zarządzania muszą działać niezależnie od stanu izolacji sieci agenta.
    5) Funkcja Naprawy (Remediate): Zatrzymuje procesy, poddaje kwarantannie pliki binarne, usuwa połączone biblioteki, usuwa pliki źródłowe i przywraca konfigurację systemu operacyjnego, aplikacji i ustawień użytkownika do stanu sprzed rozpoczęcia ataku.

6) Rollback: przywraca stan stacji końcowej do stanu z momentu utworzenia migawki VSS (Volume Shadow Copy), cofając zmiany wprowadzone przez złośliwy proces i skojarzone z nim zasoby. Agent powinien autonomicznie i w czasie zbliżonym do rzeczywistego przywrócić dane z chronionego hosta w przypadku ataku z wykorzystaniem szkodliwego oprogramowania typu ransomware .

5. Rozwiązanie EPP / EDR musi wspierać następujące modele wdrożenia :SaaS (agent-> usługa SaaS w chmurze) lub wdrożenie lokalne (urządzenie wirtualne) lub wdrożenie hybrydowe.
6. Rozwiązanie musi obsługiwać następujące mechanizmy wykrywania złośliwego oprogramowania :
   1) Przed wykonaniem (Pre-Execution): identyfikacja złośliwego oprogramowania na podstawie plików za pośrednictwem silnika reputacji. Funkcja nie wymaga aktualizacji baz danych sygnatur oraz aktualziacji plików sygnatur do realizacji swoich zadać. Dopuszcza się aby działanie tej funkcjonalności było zależne od chmury lub serwera zarządzającego – dlatego skanowanie całego dysku tym silnikiem powinno odbywać się TYLKO podczas początkowej instalacji i nie może być wymagane aby zapewnić poprawne działanie wszystkich funkcji bezpieczeństwa.
   2) Przed wykonaniem (Pre-Execution): rozwiązanie musi potrafić identyfikować nieznane szkodliwe oprogramowanie oparte na plikach na podstawie analizy statycznego z wykorzystaniem algorytmów uczenia maszynowego. Taka analiza musi odbywać się autonomicznie na stacji końcowej, bez zewnętrznych zależności lub zewnętrznego przetwarzania. Funkcjonalność nie może wymagać do działania uwzględnienia znanych IoC (DNS, IP, URL, HASH), a detekcja tego typu musi działać w czasie rzeczywistym podczas dostępu do systemu operacyjnego lub danego pliku.
   3) W czasie wykonywania (Run-Time): agent musi identyfikować i reagować na ataki z wykorzystaniem wyrafinowanych technik hackerskich (ataki bezplikowe, podatności i malware 0-day, złośliwe skrypt, lateral movement, oprogramowanie ransomware, trojany, APT itp.) Identyfikacja tych zagrożeń nie może wymagać zewnętrznych zależności, interwencji człowieka lub analizy danych poza chronioną stacją końcową. Funkcjonalność musi być realizowana w czasie zbliżonym do rzeczywistego poprzez wykorzystanie algorytmów sztucznej inteligencji. Znane IoC (DNS, IP, URL, HASH) nie mogą być wymagane jako środek identyfikacji zagrożenia.
7. Rozwiązanie musi zapewniać silny mechanizm „Anti-Tamper", czyli mechanizmy ochrony przed manipulacja oprogramowaniem przez malware lub użytkownika końcowego. Taki mechanizm musi być chroniony unikalnym hasłem dla każdego komputera końcowego. Stan WŁ./WYŁ. Ochrony przed manipulacją powinien być opcją konfigurowalną w polityce bezpieczeństwa.
8. Polityka bezpieczeństwa powinna zapewniać opcję włączenia lub wyłączenia poszczególnych silników detekcyjnych lub według typu silnika (silniki przed wykonaniem i uruchomieniem). Opcja ta nie jest wymagana dla silnika reputacji.
9. Rozwiązanie powinno zawierać otwarty interfejs API który umożliwia integracje z innymi rozwiązaniami, monitorowanie środowiska oraz automatyzacje niektórych z procesów. Dokumentacja interfejsu API powinna być natywnie dostępna z poziomu konsoli zarządzania.
10. Rozwiązanie musi obsługiwać architekturę typu Multi-Site lub Multi-Tenancy, aby całkowicie odseparować utworzone w systemie instancję i zapewnić odpowiedni dostęp administracyjny do konkretnej lokacji utworzonej zgodnie z modelem Multi-Site.

11. Rozwiązanie musi obsługiwać uwierzytelnianie SSO - SAMLv2.
12. Rozwiązanie musi obsługiwać uwierzytelnianie dwuskładnikowe (2FA) w celu uzyskania dostępu administracyjnego za pomocą aplikacji Google Authenticator lub Nano.
13. Rozwiązanie musi obsługiwać następujące formaty syslog: CEF, CEF2, RFC-5424, STIX i IOC. Rozwiązanie powinno obsługiwać certyfikaty SSL i X.509 do szyfrowania i uwierzytelniania transportu syslog.
14. Rozwiązanie musi zapewniać możliwość wysyłania wiadomości tekstowych do użytkownika stacji końcowej, bezpośrednio z konsoli zarządzania, nawet kiedy agent pracujący na stacji, znajduje się w trybie izolacji sieci / kwarantanny sieciowej.
15. Rozwiązanie musi umożliwiać zintegrowane z usługą Active Directory, aby możliwe było automatyczne przypisywanie agentów do grup, w celu powiązania ich z zasadami AD. Konsola zarządzania NIE powinna łączyć się z usługą Active Directory bezpośrednio za pośrednictwem programu ADFS ani żadnej innej metody uzyskiwania atrybutów usługi Device i User AD. Serwer zarządzania rozwiązaniem nie powinien mieć żadnych zależności od stanu usługi AD.
16. Rozwiązanie musi zawierać dashboard pokazujący wszystkie komputery, oraz możliwość ich filtrowania na podstawie atrybutów takich jak: OS, typ stacji końcowej, wersja agenta, występujące podatności, atrybuty AD, informacyjne telemetryczne, adresacja IP, charakterystyki hardware, ilości CPU, adresy Mac, interfejsy, nazwa hosta, nazwa grupy, domena). Lista powinna być dostępna do przeglądania w celu inwentaryzacji hostów, stosowania akcji dla podzbioru stacji końcowych lub mapowania stacji końcowych do grup. Musi zapewniać opcję wyświetlenia szczegółów stacji, takie jak aspekty telemetrii, stan stacji, aplikacje oraz zapewniać następujące opcje działania: Odłącz/ Połącz się od sieci (kwarantanna sieciowa, Uruchom ponownie OS, Zamknij system, Wyślij wiadomość do użytkownika, Odinstaluj agenta, Wyświetl zagrożenia.
17. Polityka ochrony stacji musi umożliwiać odpowiedź na wykryte zagrożenie w oparciu o kwalifikację zdarzenia (zagrożenie [Malicious Threat] czy podejrzane działanie [Suspicious Threat]). Odpowiedź na zagrożenie powinna umożliwiać wybranie opcje alert-only lub opcje aktywnej ochrony w oparciu o klasyfikację zagrożenia. Aktywna odpowiedź na zagrożenie, powinna być wykonywana przez autonomicznego agenta, nawet jeśli chroniona stacja nie jest podłączona do sieci.
18. Rozwiązanie EPP / EDR musi mieć zapewniać funkcjonalność lokalnego firewalla dla chronionej stacji końcowej. Ochrona firewall musi umożliwiać realizację unikalnych polityk dla każdej chronionej grupy hostów. Reguły firewalla powinny umożliwiać uwzględnienie następujących parametrów: FQDN , IP, CIDR. Funkcjonalność musi być obsługiwana dla następujących systemów operacyjnych: Windows, Linux i MacOS.
19. Rozwiązanie EPP / EDR musi mieć funkcjonalność kontroli urządzeń które próbują uzyskać dostęp do chronionej stacji. Kontrola urządzeń musi umożliwiać realizację unikalnych polityk dla każdej chronionej grupy hostów. Wymagana jest obsługa kontroli urządzeń dla następujących interfejsów: USB i Bluetooth .
20. Rozwiązanie EPP / EDR musi zarządzać podatnościami aplikacji zainstalowanych na chronionym hoście i dostarczać informacji z CVE związanych z wykrytą podatnością.
21. Przechowywanie danych EDR musi trwać co najmniej 14 dni w modelu opartym na chmurze SaaS.
22. Funkcjonalność EDR musi mieć możliwość automatycznego i autonomicznego wykonywania wstępnego indeksowania i wstępnego korelowania zdarzeń, w momencie ich

wystąpienia w chronionym środowisku. Indeksowanie powinno odbywać się w czasie rzeczywistym, a proces ten powinien odbywać się na chronionej stacji, a nie w chmurze. Powiązane ze sobą zdarzenia powinny posiadać unikalny identyfikator, który pomoże zidentyfikować grupę eventów które są ze sobą powiązane. Zapytanie zawierające tego typu identyfikator powinno zwrócić informację o wszystkich zdarzeniach (IP, DNS, PLIKI, REJESTRY, PROCESY, URL itp.) składających się na daną sytuację, niezależnie od tego czy jest ona związana ze złośliwym oprogramowaniem, czy nie. Ponadto dashboard EDR powinien zawierać eksplorator „drzewa procesów" do graficznej wizualizacji i analizy procesów które składały się na dane zdarzenie.

23. EDR musi być zintegrowany z pojedynczym Autonomicznym Agentem EPP / EDR.
24. Realizacja modułu EDR musi być zgodna z rozporządzeniem RODO.
25. EDR musi obsługiwać coustomowe reguły detekcyjne . Ta funkcjonalność ma umożliwić analitykowi przekształcenia zapytań (EDR / XDR) w automatyczne reguły detekcyjne, które wyzwalają alerty i automatyczne odpowiedzi, gdy reguły wykryją tego typu zachowanie stacji końcowej.
26. EDR musi być natywnie zintegrowany z komponentem EPP w jednym autonomicznym agencie.
27. EDR musi obsługiwać systemy operacyjne Windows, MacOS i Linux.
28. EDR musi zapewniać przeglądarkę drzewa procesów, w celu uproszczenia i automatyzacji analizy.
29. EDR musi automatycznie indeksować i korelować zdarzenia pod unikalnym identyfikatorem, który jednoznacznie wskazuje konkretne drzewo procesów.
30. Rozwiązanie EPP / EDR musi zapewniać funkcjonalność Full Remote Shell, aby administrator mógł wykonywać polecenia na stacji końcowej, nawet gdy jest ona w stanie izolacji sieciowej. Dodatkowo rozwiązanie musi zapisać transkrypcję zestawionej sesji . Taka transkrypcja musi być chroniona hasłem, a dostęp do powłoki zdalnej powinien wymuszać na Administratorze uwierzytelnianie dwuskładnikowe (2FA) w celu udzielenia dostępu. Funkcjonalność ta powinna być możliwa do włączenia / wyłączenia w polityce bezpieczeństwa rozwiązania.