Załącznik nr 1

OPIS PRZEDMIOTU ZAMÓWIENIA

CEL POSTĘPOWANIA:

Celem postępowania jest zakup klastra sprzętowego wysokiej dostępności (HA) dwóch urządzeń sieciowych o funkcjonalności load balancera polegającej na równoważeniu obciążeń i rozdzielaniu ruchu pomiędzy aplikacje/serwery w celu optymalizacji wydajności, efektywności i przepustowości infrastruktury wraz z oprogramowaniem o minimalnej funkcjonalności opisanej poniżej.

System musi realizować co najmniej następujące funkcje (posiadać moduły funkcjonalne):

- Moduł Load Balancera umożliwiający rozkład ruchu pomiędzy serwerami.
- Moduł Firewalla stanowego.
- Moduł ochrony DNS oraz globalnego równoważenia obciążenia za pomocą protokołu DNS.

SZCZEGÓŁOWE WYMAGANIA DLA DOSTARCZANYCH URZĄDZEŃ:

I. Opis wymagań:

1. Wszystkie wymienione w niniejszym dokumencie i opisane szczegółowo funkcje, muszą być dostępne w obrębie nie więcej niż jednego urządzenia fizycznego.
2. Wszystkie wymienione w niniejszym dokumencie funkcje muszą być licencjami, które nie mają ograniczeń czasowych i nie wymagają przedłużenia (są licencjami wieczystymi).
3. Zamawiający nie dopuszcza rozwiązania realizowanego w postaci maszyny (maszyn) wirtualnej w środowisku serwerowym (Vmware, Hyper-V, vcloud, KVM, Oracle VM, VirtualBox.

II. Szczegółowy opis funkcji urządzenia. Urządzenie musi realizować co najmniej następujące funkcjonalności:

1. System musi umożliwiać realizację rozkład ruchu pomiędzy serwerami:
1.1 System musi umożliwiać realizację rozdziału ruchu w oparciu o informację z warstw 4-7 modelu ISO/OSI.
1.2 System musi realizować obsługę inteligentnego równoważenia ruchu dla farm serwerów przy wsparciu dla protokołów: TCP, UDP, FTP, TFTP, http, SSL/TLS, DNS, RADIUS, SQL, RDP SSH PPTP NTP MQTT MQTT-TLS ISAKMP SMTP SIP.
1.3 System musi mieć możliwość balansowania ruchu w oparciu o algorytmy:
1.3.1 Round Robin
1.3.2 cykliczna dla usługi
1.3.3 cykliczna dla serwera
1.3.4 ważona dla usługi

1.3.5 ważona dla serwera
1.3.6 dynamiczna ważona dla usługi
1.3.7 dynamiczna ważona dla serwera
1.3.8 obserwowana
1.3.9 przewidywana
1.3.10 obciążenie serwerów
1.3.11 ilość połączeń
1.3.12 ważona ilość połączeń dla serwerów
1.3.13 ważona ilość połączeń dla usługi
1.3.14 czas odpowiedzi
1.3.15 ilość sesji
1.3.16 ważona dla ilości sesji do serwerów
1.3.17 ważona dla ilości sesji do usługi
1.3.18 hashing (URL, Domain, source IP, Destination IP)

1.4 System musi umożliwiać mechanizm dowiązania sesji (session persistence) w oparciu o:
1.4.1 cookie (hash/passive/insert lub rewrite),
1.4.2 adres źródłowy,
1.4.3 adres docelowy,
1.4.4 identyfikator sesji SSL,
1.4.5 SESSIONID,
1.4.6 SIP call,
1.4.7 Hash,
1.4.8 Host,
1.4.9 Msrdp,
1.4.10 Universal (wybranie dowolnego fragmentu payloadu i utworzenie na jego podstawie profilu persystencji.

1.5 System ma umożliwiać monitorowanie stanów serwerów i na tej podstawie dokonywania decyzji o przełączaniu w oparciu o:
1.5.1 ICMP,
1.5.2 TCP,
1.5.3 UDP,
1.5.4 HTTP,
1.5.5 HTTP/2,
1.5.6 HTTPS,
1.5.7 Inband,
1.5.8 SNMP DCA,
1.5.9 Server (co najmniej: ServerBandwidth, CPUPercentUsage, MemoryUsage, TotalClientCount),
1.5.10 Skrypty własne.

1.6 System ma wspierać content switching w oparciu o:
1.6.1 polityki: URL, URL query, URL wildcard Domain, Source IP, Destination IP, Nagłówek http, Dane HTTP i TCP, UDP,
1.6.2 protokoły w przychodzących pakietach.

1.7 System musi umożliwiać obsługę list kontroli dostępu dla 3 i 4 warstwy ISO/OSI.

1.8 System musi umożliwiać zabezpieczenia przed atakami DNS, przynajmniej takimi jak DNS query floods, DNS amplification, zabezpieczenie przed atakami SYN flood.

1.9 Selektywną kompresję danych.
1.10 Terminację sesji SSL.
1.11 Rozwiązanie musi pracować w trybie pełnego proxy.
1.12 Praca w trybie pełnego proxy nie może powodować degradacji wydajności rozwiązania.
1.13 Buforowanie połączeń TCP w przypadku osiągnięcia zadanej ilości sesji dla danego serwera.
1.14 Wsparcie dla HTTP/2, w tym wsparcie dla kompresji nagłówków.
1.15 System musi posiadać funkcję definiowania maksymalnej ilości obsługiwanych przez dany serwer połączeń, w przypadku przekroczenia zdefiniowanej wartości musi istnieć możliwość wysłania klientowi strony błędu lub przekierowania klienta na inny serwer.
1.16 System musi zapewniać możliwość klonowania puli serwerów umożliwiając wysyłanie kopii ruchu do zewnętrznych systemów monitoringu lub urządzeń typu IDS/IPS.
1.17 System musi umożliwiać obsługę sieci co najmniej w zakresie: Routingu statycznego, Routingu dynamicznego OSPF, RIP, BGP, IS-IS, Link Aggregation, 802.3ad, VLAN 802.1q Jumbo Frames, z wykorzystaniem integracji z Equal Cost Multi-Path (ECMP).
1.18 System musi wspierać wysoką dostępność: Active/passive Active/active Connection Mirroring.
1.19 Urządzenie powinno implementować TCP proxy z mechanizmem zamykania okna w stronę serwera www w przypadku zbyt wolnego odbierania danych przez zdalnego klienta.
1.20 Urządzenie musi mieć możliwość strojenia profili połączeń TCP, w tym co najmniej:
1.20.1 Ustawienia timeout idle,
1.20.2 keep alive interval,
1.20.3 zero window timeout,
1.20.4 Initial Receive Window Size,
1.20.5 Wyboru algorytmu wykrywania natłoku,
1.20.6 Wsparcia hybrydowych algorytmów wykrywania natłoku, opartych o stratę pakietów i round-trip-time,
1.20.7 Mechanizmu rate pacing,
1.20.8 określenia maksymalnego wielkości bufora per połączenie TCP celem ochrony przed wysyceniem pamięci,
1.20.9 określenia maksymalnego bufora wysyłania i odbioru (send/receive buffer).
1.21 Urządzenie musi mieć możliwość włączenia ignorowania nagłówków przeglądarki dotyczących cachowania (Cache-control).
1.22 Urządzenie musi umożliwiać selektywne cachowanie zwracanego kontentu a cachowanie powinno być zależne od:
1.22.1 Listy URN-ów i prefixów URN-ów z których zwracany jest kontent,
1.22.2 Maksymalnej wielkości zwracanego obiektu,
1.22.3 minimalnej wielkości zwracanego obiektu.
1.23 Powinna istnieć możliwość ograniczania wielkości cachu, ilości cachowanych obiektów celem ochrony pamięci urządzenia przed przepełnieniem.
1.24 Urządzenie musi wspierać multipleksacje wielu zapytań http w tej samej sesji TCP.

1.25 Urządzenie musi umożliwiać kompresję zwracanej zawartości http. Użycie kompresji powinno być zależne od:
- 1.25.1 Listy dozwolonych URI,
- 1.25.2 Listy wykluczonych URI,
- 1.25.3 Listy możliwych do kompresji Content-Type,
- 1.25.4 Listy wykluczonych Content-Type,
- 1.25.5 Minimalnej wielkości zwracanego obiektu.

1.26 Kierowanie ruchu do odpowiedniej pooli zasobów:
- 1.26.1 Urządzenie powinno realizować mechanizm kierowania żądań od użytkowników końcowych do pool (grup) zasobów serwerowych na podstawie następujących atrybutów:
  - 1.26.1.1 Wartości http hostname, początku/końca nagłówka hostname, ciąg znaków znajdujący się w nagłówku hostname,
  - 1.26.1.2 port,
  - 1.26.1.3 metody http,
  - 1.26.1.4 nazwy cookie, ciągu znaków na początku/końcu nazwy cookie, ciągu znaku zawartego w nazwie cookie,
  - 1.26.1.5 URN-a, oraz ciągu znaków występujących na początku/końcu lub zawartych w URN-ie,
  - 1.26.1.6 User agent,
  - 1.26.1.7 Dowolnego nagłówka lub kombinacji nagłówków,
  - 1.26.1.8 Wykorzystania cipherów w TLS,
  - 1.26.1.9 Common Name Klienta, oraz ciągu znaków występujących na początku/końcu lub zawartych w Common Name.
- 1.26.2 Reguły kierowania żądań powinny wymuszać co najmniej następujące akcje:
  - 1.26.2.1 Wybór pooli zasobów (grupy IP adresów i portów),
  - 1.26.2.2 Kierowania żądania do konkretnego adresu IP,
  - 1.26.2.3 Redyrekcji,
  - 1.26.2.4 Wyłączanie / włączanie ochrony aplikacyjnej,
  - 1.26.2.5 Wybór polityki ochrony aplikacyjnej,
  - 1.26.2.6 Włączania/ wyłączania cachowania,
  - 1.26.2.7 Włączania/ wyłączania kompresji,
  - 1.26.2.8 Dodania / usunięcia nagłówka / cookie http,
  - 1.26.2.9 Wyboru parametrów profilu (zestawu parametrów, w tym CA, certyfikatu, klucza) dla połączeń w stronę serwerów aplikacyjnych.
- 1.26.3 Powinna istnieć możliwość nadpisania statycznych reguł kierowania ruchu w języku skryptowym, na podstawie dowolnych atrybutów żądania http, oraz stanu sesji zapisanego w pamięci urządzenia.

1.27 System musi zapewniać obsługę certyfikatów z kluczami typu ECDSA wykorzystującymi krzywe eliptyczne (ECC) zarówno od strony klienta, jak i od strony puli serwerów,

1.28 Sprzętowe wsparcie dla algorytmów AES, AES-GCM, RSA, DSA, DH, ECDSA, ECDH, SHA2. Wsparcie dla Perfect Forward Secrecy.

1.29 Dla protokołu TLS 1.2 wymagana jest obsługa AES-GCM zarówno od strony klienta, jak i od strony puli serwerów.

1.30 Wsparcie dla protokołu TLS 1.3.

1.31 System musi zapewniać obsługę certyfikatów podpisanych funkcją skrótu SHA-2 zarówno od strony klienta, jak i od strony puli serwerów.

1.32 System musi posiadać funkcję walidacji certyfikatów klientów łączących się przy wykorzystaniu protokołu SSL.
1.33 System musi obsługiwać sieci VLAN w standardzie 802.1q.
1.34 System musi obsługiwać agregację linków w standardzie 802.3ad (LACP).
1.35 System musi obsługiwać Jumbo Frames.
1.36 System musi posiadać funkcjonalność bramy VXLAN oraz NVGRE.
1.37 Klucze prywatne zapisane na dysku urządzenia muszą być zaszyfrowane. Nie dopuszcza się rozwiązań przechowujących klucze prywatne w formie jawnej.
1.38 Rozwiązanie musi posiadać wbudowany w system operacyjny język skryptowy, posiadający co najmniej następujące cechy:
1.38.1 Analiza, zmiana oraz zastępowanie parametrów w nagłówku http oraz w zawartości pakietów,
1.38.2 Obsługa (parsowanie, modyfikacja, kreowanie odpowiedzi) protokołów: http, http2, WebSocket, DNS, RADIUS,
1.38.3 Język skryptowy musi umożliwiać dynamiczne przypisanie polityki bezpieczeństwa aplikacyjnego w oparciu o dowolną kombinację atrybutów http, SSL/TLS. Parsowanie parametrów komunikacji TLS/SSL, w szczególności odczyt certyfikatów X.509 i wstrzykiwanie odczytanych parametrów w formie nagłówków http,
1.38.4 Możliwość zdefiniowania reakcji na zagrożenie wykryte przez Web Application Firewall na podstawie danych dostarczonych przez WAF, odczytanych w komunikacji http, SSL, WebSocket oraz historycznych atrybutów konfiguracji (kontekstu).
1.38.5 Język skryptowy musi być oparty o Tool Command Language i mieć możliwość delegowania złożonych zadań do silnika NodeJS oraz użycia bibliotek „społecznościowych”, celem np. parsowania protokołów jeszcze nie zaimplementowanych w rozwiązaniu Application Delivery Controller.

2. System musi umożliwiać realizację firewalla stanowego:
2.1 Konfigurację reguł firewallowych za pomocą wbudowanego w system interfejsu graficznego.
2.2 Rozwiązanie powinno śledzić i ograniczać wystąpienia anomalii warstw niższych, w tym: IPv4, IPv6, TCP, UDP, ICP.
2.3 Rozwiązanie powinno zawierać system sztucznej inteligencji analizujący na bieżąco ruchu i wykrywające ilościowe anomalie ruchu.
2.4 Rozwiązanie powinno zawierać w pełni funkcjonalny firewall warstwy 4.
2.5 Rozwiązanie powinno być wyposażone w system ochrony przed atakami DDoS realizowany przynajmniej częściowo za pomocą dedykowanych komponentów hardware.
2.5.1 Rozwiązanie powinno obsłużyć sprzętowo (w hardwarze) mitygację minimum 100 wektorów ataków DDoS.
2.6 Rozwiązanie powinno umożliwiać uruchomienie proxy SSH, które umożliwia np. blokowanie ściągania lub wgrywania plików po SCP lub SFTP, ustawienie czy użytkownik ma dostęp do shella.

3. System ochrony DNS musi realizować co najmniej następujące funkcjonalności:

3.1.1 DNS Autorytatywny,
3.1.2 Global Server Load Balancing,
3.1.3 Dynamiczne podpisywanie domen DNS SEC,
3.1.4 DNS Caching,
3.1.5 DNS Resolver,
3.1.6 Walidacja podpisów odpowiedzi DNS SEC,
3.1.7 Rozkład ruchu pomiędzy serwerami DNS innych producentów,
3.1.8 Ochrona przed atakami DDoS na protokół DNS,
3.1.9 Ochrona przed atakami DDoS na niższe (niż DNS) protokoły w modelu ISO/OSI,

3.2 System musi natywnie wspierać IPv4 i IPv6.
3.3 System musi obsługiwać sieci VLAN w standardzie 802.1q.
3.4 System musi obsługiwać agregację linków w standardzie 802.3ad (LACP).
3.5 Rozwiązanie powinno wspierać rozrzucanie ruchu DNS na podstawie następujących algorytmów:
3.5.1 cykliczna,
3.5.2 ważona,
3.5.3 najmniejsza liczba połączeń,
3.5.4 najszybsza odpowiedź serwera,
3.5.5 najmniejsza liczba połączeń i najszybsza odpowiedź serwera,
3.5.6 najmniejsza liczba połączeń i najszybsza odpowiedź serwera w zdefiniowanym czasie,
3.5.7 dynamicznie ważona oparta na SNMP/WMI,
3.5.8 definiowana na podstawie grupy priorytetów dla serwerów,
3.5.9 musi istnieć możliwość modyfikacji metod równoważenia obciążenia pomiędzy serwerami przy wykorzystaniu wbudowanego języka skryptowego.
3.6 Rozwiązanie musi posiadać wbudowany w system operacyjny język skryptowy, posiadający co najmniej następujące cechy:
3.6.1 Analiza, zmiana oraz zastępowanie parametrów DNS,
3.6.2 Generowania odpowiedzi na podstawie zapytania DNS oraz stanu zapisanego w pamięci urządzenia (np. historyczne zapytania DNS od danego użytkownika),
3.6.3 Powinna istnieć możliwość nadpisania statycznych reguł kierowania ruchu, na podstawie dowolnych atrybutów żądania DNS, oraz stanu sesji zapisanego w pamięci urządzenia,
3.6.4 Język skryptowy powinien umożliwiać iteracje poprzez wszystkie rekordy odpowiedzi DNS, oraz odczytywanie, modyfikacje i usuwanie poszczególnych rekordów,
3.6.5 Język skryptowy powinien umożliwiać odczyt, modyfikacje i wstawienie sekcji additional,
3.6.6 Język skryptowy powinien umożliwiać odczyt i modyfikację:
3.6.6.1 Typu rekordu,
3.6.6.2 TTL,
3.6.6.3 Nazwy rekordu,
3.6.6.4 Klasy (class).
3.6.7 Język skryptowy powinien umożliwiać operacje porównania stringów: equals, contain, starts with, ends with,

3.6.8 Język skryptowy powinien wspierać REGEXP,
3.6.9 Język skryptowy powinien umożliwiać selektywne (per request) wyłączanie funkcji GSLB, DNS Resolver i Cache, autorytatywnego DNS,
3.6.10 Język skryptowy powinien umożliwiać selektywne (per request) wyłączanie funkcji GSLB, DNS Cache.

3.7 Powinna istnieć możliwość weryfikacji listy rekordów i reakcji / obsługi wyjątków, jeżeli domena lub zwracany adres IP znajduje się na liście.

3.8 Powinna istnieć możliwość zaimportowania i użyciu statycznej listy „wyjątków" powyżej 200 000 rekordów.

3.9 Powinna istnieć możliwość definiowania obsługi ruchu zarówno dla żądań jak i odpowiedzi DNS.

3.10 Rozwiązanie powinno pełnić funkcje serwera autorytatywnego.

3.11 Celem podwyższenia wydajności, zony statyczne powinny być ładowane do pomięci i obsługiwane z pamięci RAM.

3.12 Autorytatywny serwer DNS powinien działać zarówno jako primary/master jak i secondary/slave.

3.13 Rozwiązanie powinno wspierać zone transfer
3.13.1 obsługiwać TSIG,
3.13.2 inicjować transfer zony po otrzymaniu notyfikacji (NOTIFY),
3.13.3 Powinna istnieć możliwość przekazania notyfikacji do innych serwerów DNS.

3.14 Powinna istnieć możliwość definiowania nazw dla których jest implementowany mechanizm GSLB (Global Server Load Balancing).

3.15 Nazwy zwracające dynamiczne rekordy (GSLB) powinny nadpisywać statyczną definicje strefy (zony).

3.16 Mechanizm GSLB powinien obsługiwać następujące typy rekordów:
3.16.1 A,
3.16.2 AAAA,
3.16.3 CNAME,
3.16.4 MX,
3.16.5 NAPTR,
3.16.6 SRV.

3.17 Silnik GSLB powinien zwracać adresy IP do zasobów, które obsługują aplikacje z zapytania DNS. Adresy IP będą dalej nazywane serwerami virtualnymi.

3.18 Silnik GSLB powinien uwzględniać stan zasobów znajdujących się za adresami IP, do których może być skierowany użytkownik poprzez zwrócenie adresu IPv4 lub IPv6.

3.19 Stan zasobów powinien być wymieniany asynchronicznie pomiędzy:
3.19.1 Innymi autorytatywnymi serwerami DNS implementującymi GSLB,
3.19.2 Urządzeniami ADC (Application Delivery Controller) pochodzącymi od tego samego producenta,

3.20 Rozwiązanie powinno wspierać mechanizm Route Healt Injection (RHI),

3.21 Rozwiązanie GSLB powinno wspierać co najmniej następujące metody monitorowania:
3.21.1 http, https,
3.21.2 IMAP,
3.21.3 LDAP,
3.21.4 MSSQL, MySQL, PostgresSQL,
3.21.5 NNTP,
3.21.6 POP3,
3.21.7 RADIUS i RADIUS Accounting,
3.21.8 SIP,

3.21.9 SNMP,
3.21.10 SMTP,
3.21.11 SOAP,
3.21.12 TCP,
3.21.13 UDP,
3.21.14 Monitora skryptowego, definiującego sekwencje wymiany wiadomości
3.21.15 Monitora zewnętrznego, czyli uruchomienia procesu np. bash, perl, zwracającego stan zasobów na podstawie logiki zawartej w zewnętrznym programie / skrypcie.

3.22 Silnik GSLB powinien wspierać co najmniej następujące algorytmy load balancingu:
3.22.1 Cykliczna,
3.22.2 Ważona,
3.22.3 Na podstawie adresu IP resolvera, implementująca persystencji (przywiązywanie sesji),
3.22.4 Do pierwszego dostępnego virtualnego serwera, przy czym rozwiązanie powinno umożliwiać zdefiniowanie kolejności virtualnych serwerów,
3.22.5 Lokalizacji pytającego serwera, na podstawie zaszytej bazy geolokacyjnej.

3.23 Jeżeli wirtualny serwer jest zdefiniowany na urządzeniu ADC od tego samego producenta, silnik GSLB powinien wspierać co najmniej następujące algorytmy load balancingu:
3.23.1 Na podstawie wolnych zasobów procesora urządzenia ADC,
3.23.2 Ilości hopów pomiędzy LDNS a ADC,
3.23.3 Na podstawie aktualnego wysycenia połączeń do serwera wirtualnego, Mniejsze wysycenie oznacza większe prawdopodobieństwo wyboru,
3.23.4 Aktualnej ilości obsługiwanych połączeń przez ADC,
3.23.5 Pakietów na sekundę,
3.23.6 Round Trip Time,
3.23.7 Subiektywnej oceny wydajności przypisanej do wirtualnego serwera na urządzeniu ADC przez administratora,
3.23.8 Średniej ważonej z powyższych metryk.

3.24 Powinna istnieć możliwość grupowania obszarów geograficznych z dokładnością do poziomu województwa, tak aby zwiększyć prawdopodobieństwo połączenia Klienta z najbliższym CPD.

3.25 Powinna istnieć możliwość grupowania prywatnych adresacji w obszary i wykorzystania tych obszarów do algorytmu load balancingu.

3.26 Silnik GSLB powinien umożliwiać grupowanie zasobów / wirtualnych serwerów w Centra Przetwarzanie danych na podstawie ich rzeczywistej lokalizacji.

3.27 System GSLB powinien wspierać mechanizmy oceny dostępności Centrum Przetwarzania Danych.

3.28 W przypadku niedostępności Centrum Przetwarzania Danych, GSLB nie powinno zwracać adresów IP posadowionych w niedostępnym CPD, chyba że są one zdefiniowane statycznie.

3.29 Powinna istnieć możliwość delegacji procesu monitorowania wirtualnych serwerów posadowionych na rozwiązaniach firm trzecich do urządzenia ADC od tego samego dostawcy znajdującego się w tym samym CPD co wirtualne serwery.

3.30 Powinna istnieć możliwość delegacji procesu monitorowania wirtualnych serwerów, posadowionych na rozwiązaniach firm trzecich, do urządzenia ADC mającego routing do strefy, w której znajdują się monitorowane zasoby.

3.31 Powinna istnieć możliwość grupowania nazw, tak żeby dany klient był obsługiwany z tego centrum przetwarzania danych dla wszystkich nazw ze zdefiniowanego zbioru.

3.32 Rozwiązanie powinno wspierać trzy tryby pracy:

3.32.1 Cache transparentny, gdzie cachowane są pełne odpowiedzi. W przypadku potrzeby rozwiązania zapytania (cache miss), cache transparentny przesyła zapytanie do drugiego poziomu cache-u,
3.32.2 Resolvera z cachem,
3.32.3 Resolvera walidującego podpisy DNS SEC z cachem.

3.33 Rozwiązanie powinno umożliwiać granularną konfigurację każdego typu cache'u, tak aby uzyskać optymalny stosunek czasu wyszukiwania rekordów do cache hit ratio. Rozwiązanie powinno umożliwiać strojenie przynajmniej następujących parametrów
3.33.1 Dla cache-u transparentnego: dostępnej pamięci dla cachu wiadomości i cachu rekordów,
3.33.2 W trybie pracy resolver z cachem: dostępnej pamięci dla cachu wiadomości, cachu rekordów i cachu autorytatywnych serwerów nazw,
3.33.3 W trybie pracy resolver walidującego DNS SEC z cachem: dostępnej pamięci dla cachu wiadomości, cachu rekordów, cachu autorytatywnych serwerów nazw i cachu kluczy DNS SEC.

3.34 Resolver powinien zbierać i utrzymywać statystyki czasu odpowiedzi serwerów autoratywnych i wykorzystywać te statystyki w celu skrócenia czasu rozwiązywania nazw.

3.35 Rozwiązanie powinno wykorzystywać IPv4 i IPv6 do komunikacji z serwerami autorytatywnymi.

3.36 Rozwiązanie powinno umożliwiać przeszukiwanie cachowanych rekordów.

3.37 Rozwiązanie powinno umożliwiać selektywne usuwanie cachowanych rekordów.

3.38 System powinien filtrować w oparciu o Response Policy Zone.

3.39 System powinien filtrować w oparciu o pliki płaskie, gdzie w pliku płaskim do każdej domeny może być przypisana akcja, np. NX Domain, odpowiedź CNAME, A record itd.

3.40 Resolver powinien zbierać co najmniej następujące statystyki:
3.40.1 Ilość zapytań,
3.40.2 Odpowiedzi,
3.40.3 Wykorzystanych autorytatywnych serwerów,
3.40.4 Odpowiedzi wygenerowanych lokalnie,
3.40.5 Trafień w Response Policy Zone,
3.40.6 Trafień / nietrafień w cache, per typ cachu (cache wiadomości, resource rekord, autorytatywne servery),
3.40.7 Wyników rozwiązywań zapytań, np. sukces, timeout, przepełnienie kolejki,
3.40.8 Odpowiedzi ze strony serwerów autorytatywnych.

3.41 Statystyki powinny być dostępne za pomocą SNMP.

3.42 Powinna istnieć możliwość definiowania wyjątków dla zapytań trafiających do systemu resolvera, przynajmniej dla następujących przypadków:
3.42.1 Powinna istnieć możliwość przekazania zapytania dla predefiniowanych subdomen do pooli serwerów DNS odpowiedzialnych za te domeny,
3.42.2 Powinna istnieć możliwość przekazania zapytania dla predefiniowanych domen odwrotnych do pooli serwerów DNS odpowiedzialnych za te domeny odwrotne,
3.42.3 Generowania predefiniowanej odpowiedzi, jeżeli w oryginalnej odpowiedzi z serwera autorytatywnego znajdzie się IP adres o złej reputacji.

3.43 Silnik systemu (resolvera) powinien mieć możliwość skorzystania z bazy reputacyjnej adresów IP celem usuwania tychże adresów z odpowiedzi od serwera autorytatywnego.

3.44 Rozwiązanie powinno filtrować po dowolnej kombinacji następujących typów zapytań:
a, axfr, dnskey, isdn, mb, naptr, nsec3param, rrsig, sshfp , a6, caa, ds, ixfr, md, nimloc, null, rt, tkey , aaaa, cert, eid, key, mf, ns, nxt, sig, tsig , afsdb, cname, gpos,

kx, mg, nsap, opt, sink, txt , any, dhcid, hinfo, loc, minfo, nsap-ptr, ptr, soa, wks , apl, dlv, hip, maila, mr, nsec, px, spf, x25 , atma, dname, ipseckey, mailb, mx, nsec3, rp, srv, zxfr.

3.45 Rozwiązanie powinno być odporne na ataki typu „Phantom domain attack”.

3.46 Rozwiązanie powinno być odporne na ataki typu Lock-up domain”.

3.47 Rozwiązanie powinno śledzić i ograniczać

3.47.1 Liczbę pakietów o nieprawidłowej składni,

3.47.2 Liczbę pakietów per typ zapytania: A, PTR, NS, SOA, CNAME, MX, AAAA, TXT, SRV, AXFR, IXFR, ANY, NXDOMAIN, niemieszczących się w powyższych kategoriach,

3.47.3 Występowanie ataków typu reflection.

3.48 System musi posiadać moduł analizy ruchu DNS. Moduł powinien zbierać następujące metryki:

3.48.1 Typy zapytań,

3.48.2 Domeny,

3.48.3 Najaktywniejsi Klienci per IP, Kraj,

3.48.4 Typy ataków,

3.48.5 Wektory ataków.

3.49 Dla funkcjonalność GSLB system musi realizować co najmniej następujące funkcjonalności:

3.49.1 Global Server Load Balancing,

3.49.2 Dynamiczne podpisywanie domen DNS SEC,

3.49.3 DNS Caching,

3.49.4 Walidacja podpisów odpowiedzi DNS SEC,

3.49.5 Rozkład ruchu pomiędzy serwerami DNS innych producentów,

3.49.6 Ochrona przed atakami DDoS na protokół DNS,

3.49.7 Ochrona przed atakami DDoS na niższe (niż DNS) protokoły w modelu ISO/OSI.

3.50 Rozwiązanie powinno wspierać cache transparentny, gdzie cachowane są pełne odpowiedzi. W przypadku potrzeby rozwiązania zapytania (cache miss), cache transparentny przesyła zapytanie do drugiego poziomu cache-u.

3.51 Rozwiązanie powinno umożliwiać granularną konfigurację cache’u, tak aby uzyskać optymalny stosunek czasu wyszukiwania rekordów do cache hit ratio. Rozwiązanie powinno umożliwiać strojenie dostępnej pamięci dla cachu wiadomości i cachu rekordów.

4. Wysoka dostępność i analityka

4.1 System musi posiadać co najmniej następujące interfejsy administracyjne:

4.1.1 GUI przy wykorzystaniu protokołu https

4.1.2 Zarządzanie poprzez SSH

4.1.3 Zarządzanie poprzez API REST

4.2 System musi posiadać moduł analizy ruchu http. Moduł powinien zbierać następujące metryki

4.2.1 Czas odpowiedzi per serwer

4.2.2 Czas odpowiedzi per URI

4.2.3 Ilość sesji użytkownika

4.2.4 Przepustowość

4.2.5 Adres źródła

4.2.6 Kraj
4.2.7 User Agent (wykorzystywana przez klienta aplikacja)
4.2.8 Metoda dostępu

4.3 System musi posiadać następujące funkcje zarządzania siecią:
4.3.1 Obsługa protokołu SNMP v1/v2c/v3
4.3.2 Możliwość budowania własnych zdarzeń SNMP z własnymi numerami OID
4.3.3 Zewnętrzny syslog
4.3.4 Możliwość wysyłania logów syslog do więcej niż jednego miejsca docelowego
4.3.5 Zbieranie danych i ich wyświetlanie
4.3.6 Zbieranie danych zgodnie z ustawieniami administratora
4.3.7 Osobna brama domyślna dla interfejsu zarządzającego
4.3.8 Wsparcie dla przynajmniej 2 wersji oprogramowania (multi-boot)
4.3.9 zapisywanie konfiguracji (możliwość szyfrowania i eksportu kluczy)
4.3.10 Dedykowany podsystem monitorowania stanu pracy urządzenia (always on management) z funkcjami restartu, wstrzymania oraz sprzętowego resetu systemu.

4.4 System musi umożliwiać weryfikację działającego na urządzeniu firmware, czy nie uległ on modyfikacji (TPM Chain of Custody).

4.5 System musi posiadać funkcję integracji z zewnętrznymi serwerami uwierzytelnienia użytkowników LDAP, RADIUS, TACACS.

4.6 Autoryzacja administratorów systemu musi bazować na rolach użytkowników.

4.7 System musi posiadać funkcję definiowania i edycji szablonów konfiguracji aplikacji. Szablony powinny służyć do optymalizacji procesu wdrażania systemu zarówno dla znanych aplikacji biznesowych, jak i własnych aplikacji klienta.

4.8 Rozwiązanie musi oferować podział na tzw. partycje administracyjne. Zdefiniowany użytkownik może zarządzać konfiguracją tylko i wyłącznie wewnątrz swojej partycji.

4.9 Rozwiązanie musi oferować wsparcie dla tzw. domen routingu (Virtual Routing and Forwarding). Rozwiązanie takie oferuje separację ruchu sieciowego do różnych aplikacji. Musi umożliwiać poprawnie działanie rozwiązania, kiedy podłączone VLANy do urządzenia mają takie same podsieci i adresy IP.

4.10 Rozwiązanie musi oferować stworzenie minimum 1000 partycji administracyjnych oraz 1000 jednoczesnych domen routingu. Partycje administracyjne i domeny routingu muszą być dostępne również, jeżeli urządzenie pracuje w formie klastra.

4.11 Klaster wysokiej dostępności (HA) umożliwiający pracę w trybie active – standby powinien mieć możliwość realizacji również trybu active-active oraz rozbudowy do klastra N+1.

4.12 W ramach klastra musi istnieć możliwość jednoczesnego wykorzystania różnych modeli urządzeń sprzętowych oraz maszyn wirtualnych.

4.13 Klaster wysokiej dostępności musi zapewniać kopiowanie informacji o sesji SSL i stanu sesji TCP pomiędzy urządzeniami, aby uniknąć ponownej negocjacji po przełączeniu ruchu.

4.14 Klaster wysokiej dostępności musi zapewniać synchronizację:
4.14.1 Konfiguracji
4.14.2 Stanu połączeń

4.14.3 Przywiązywania sesji (Session persistence)
4.14.4 Polityk bezpieczeństwa aplikacyjnego

4.15 Wykrycie awarii urządzeń w klastrze odbywać się musi przy użyciu, co najmniej następujących metod:

4.15.1 Weryfikacja stanu pracy urządzenia poprzez analizę aktywności w sieci (Network failover)

GWARANCJA I SERWIS:

Każde z urządzeń dostarczone przez Zamawiającego musi być objęte 36 miesięczną gwarancją producenta. W obrębie gwarancji zawarte musi być:

1.1 Dostęp do aktualnych wersji oprogramowania oraz dokumentacji producenta.
1.2 Sposób obsługi zgłoszeń gwarancyjnych w trybie 5x10 (dni x godzin).
1.3 Wymiana sprzętu następnego dnia roboczego po identyfikacji usterki.
1.4 Aktualizacja oprogramowania dostarczonych urządzeń przez Wykonawcę.

USŁUGA DWROŻENIA U ZAMAWIAJĄCEGO

1. W ramach usługi wdrożenia Wykonawca dokona przeglądu konfiguracji urządzeń F5 serii i5800 posiadanych przez Zamawiającego, sporządzi projekt wymiany posiadanych przez Zamawiającego urządzeń na dostarczone przez Zamawiającego. Sporządzenie projektu będzie polegać na przeanalizowaniu konfiguracji posiadanych przez Zamawiającego urządzeń F5 serii i5800 wraz z konfiguracją urządzeń, które będą podłączone do dostarczonych urządzenia pod kątem poprawności konfiguracji, zgodności ze standardami obecnie stosowanymi przez producenta urządzeń. Wszystkie prace konfiguracyjne zostaną przeprowadzone w terminie ustalonym z Zamawiającym;
2. Zamawiającym po podpisaniu Umowy udostępni aktualną konfigurację posiadanych urządzeń sieciowych będących częścią wdrożenia a także pozostałych urządzeń których konfiguracja będzie niezbędna do poprawnego wykonania wdrożenia.
3. Wykonawca wykona usługę instalacji i konfiguracji dostarczonych Urządzeń i oprogramowania w miejsce posiadanych przez Zamawiającego w/w urządzeń firmy F5 serii i5800 w budynku O1 Polskiego Radia przy al. Niepodległości 77/85.
4. Instalacja musi być zaplanowana w sposób, który umożliwi jak najkrótszy czas niedostępności istniejącej sieci Zamawiającego.
5. Instalacja nastąpi po zaakceptowaniu przez Zamawiającego projektu migracji wykonanego przez Wykonawcę. Po zakończeniu prac, Wykonawca sporządzi dokumentację powykonawczą zawierającą aktualną konfigurację urządzeń wraz z opisem i rysunkami odwzorowującymi aktualną strukturę sieci.

6. Prace będą wykonywane na teranie Warszawy w godzinach uzgodnionych z Zamawiającym.

USŁUGA SZKOLENIA

1. Zamawiający wymaga voucherów na szkolenia dla trzech osób z zakresu zaoferowanych urządzeń i oprogramowania.
2. Szkolenia muszą odbyć się na terenie Polski, w certyfikowanym ośrodku producenta dostarczonych Urządzeń i Oprogramowania w ciągu 12 miesięcy od daty podpisania protokołu odbioru technicznego. Zamawiający dopuszcza formę szkolenia on-line (na odległość).
3. Zakresem szkolenia musi być architektura działania dostarczonych Urządzeń i sposób ich konfiguracji.
4. Szkolenia muszą być prowadzone przez instruktora w języku polskim.