İSTANBUL TİCARET ODASI

1882

İSTANBUL TİCARET ODASI

WEB UYGULAMA GÜVENLİK DUVARI & YÜK DENGELEME CİHAZI TEKNİK ŞARTNAMESİ

2025

İÇİNDEKİLER

İSTANBUL TİCARET ODASI WEB UYGULAMA GÜVENLİK DUVARI & YÜK DENGELEME HİZMETİ TEKNİK ŞARTNAMESİ

1. GENEL ŞARTLAR VE YÜKLENİCİ YÜKÜMLÜLÜKLERİ

1.1. Bu şartname, İstanbul Ticaret Odası'nın web tabanlı uygulamalarının güvenliğinin sağlanması, erişim sürekliliğinin korunması ve tehditlere karşı merkezi yönetimle savunma sağlanması amacıyla Web Uygulama Güvenlik Duvarı (WAF) ve Yük Dengeleme (Load Balancer) çözümlerinin teminini ve sanal ortamda kurulmasını kapsar.

1.2. Teklif edilecek çözüm, üreticisinin desteklediği sanallaştırma platformlarında (örneğin VMware ESXi, KVM, Microsoft Hyper-V vb.) çalışabilecek şekilde sanal appliance (Virtual Appliance) olarak sağlanmalı ve kurumun mevcut altyapısına uyumlu olacak şekilde konumlandırılmalıdır.

1.3. Fiziksel donanım, cloud-native hizmet veya bulut servis modeli sunulması bu proje kapsamında kabul edilmeyecek; yalnızca sanal platformlarda çalışabilen lisanslı yazılım çözümleri değerlendirilecektir.

1.4. Çözüm, yüksek erişilebilirlik (High Availability), yük devretme (Failover), merkezi yönetim ve detaylı loglama gibi kurumsal seviyede özellikler sunmalıdır.

1.5. Yüklenici, kurulacak sistemin tüm bileşenlerinin kesintisiz, güvenli ve tam performansla çalışmasını sağlayacak lisans, teknik dokümantasyon ve yapılandırmaları eksiksiz biçimde temin etmekle yükümlüdür.

1.6. Kurulum sonrası sistemin en az 3 yıl süreyle üretici desteği kapsamında olması, yazılım güncellemeleri ve güvenlik imzalarının sürekli sağlanması gerekmektedir. Lisanslar, sistem kurulumunun tamamlandığı gün itibarıyla aktif edilecektir.

1.7. Yüklenici, sistemin devreye alınmasını takiben gerekli eğitimleri verecek, dokümantasyon sağlayacak ve kurulumla ilgili tüm yapılandırmaları İstanbul Ticaret Odası yetkililerinin erişimine açık şekilde gerçekleştirecektir.

1.8. Teklif edilen çözümün teknik şartnameye uygunluğu, üretici belgeleri veya teknik kataloglarla desteklenmelidir.

1.9. Teklif edilecek Web Uygulama Güvenlik Duvarı (WAF) çözümü, uluslararası bağımsız analiz firmalarının (örneğin Gartner, Forrester, IDC) son 3 yıl içerisinde yayımladığı "Magic Quadrant", "Wave Report" veya eşdeğer teknik değerlendirme raporlarında **lider** veya **vizyoner** kategorilerde konumlandırılmış bir üreticiye ait olmalıdır.

1.10. Yüklenici, bu teknik şartnamede belirtilen tüm maddelere ve teknik gereksinimlere **eksiksiz** şekilde uymakla yükümlüdür. Teklif edilen ürün veya hizmetlerin, şartnamede tanımlanan koşulları doğrudan karşılaması esastır; aksi durumlar teklifin reddedilmesi veya sözleşmenin feshiyle sonuçlanabilir.

İSTANBUL TİCARET ODASI WEB UYGULAMA
GÜVENLİK DUVARI & YÜK DENGELEME HİZMETİ
TEKNİK ŞARTNAMESİ

1.11. Bu şartname kapsamında teklif edilecek ürün, lisans, bakım ve eğitim hizmetlerine ilişkin mali ve teknik tekliflerin ayrıntılı olarak sunulabilmesi amacıyla EK-1'de Teklif Tablosu hazırlanmıştır. Teklif sahipleri, bu tabloyu eksiksiz doldurmakla yükümlüdür.

EK -1: TEKLİF TABLOSU

Kalem No	Kalem Türü	Model / Kod	Ürün veya Hizmet Açıklaması	Miktar	Birim	Süre (Yıl / Gün)	Birim Fiyat USD/EUR/TL (KDV Hariç)	Genel Toplam (KDV Hariç)	Şartnameye Uygunluk (E/H)
1	Ürün/ Lisans				Adet				
2	Bakım/ Destek				Yıl				
3	Eğitim				Gün				
Toplam Fiyat									

2. WEB UYGULAMA GÜVENLİK DUVARI İÇİN GENEL ŞARTLAR

2.1. Sistem, sanal appliance mimarisi ile teklif edilecektir. Teklif edilen çözüm, üreticisinin desteklediği sanallaştırma platformlarında (örneğin VMware ESXi, KVM, Microsoft Hyper-V, VirtualBox vb.) çalışabilir nitelikte olmalıdır.

2.2. Sistemin tüm bileşenleri için en az **3 yıl süreli lisans** sağlanmalı, bu süre boyunca ürün tüm özellikleriyle tam fonksiyonel olarak çalışmalıdır. Kullanıcı, trafik veya oturum sayısıyla sınırlı olmayan lisans yapısı tercih edilmelidir.

2.3. Web Uygulama Güvenlik Duvarı çözümünün Layer 7 (Uygulama Katmanı) throughput kapasitesi **en az 1.0 Gbps** olmalıdır.

2.4. HTTP işlem kapasitesi **en az 50.000 TPS (Transaction Per Second)**, SSL işlem kapasitesi ise **en az 15.000 TPS** olmalıdır.

2.5. Sistem, **en az 8 vCPU** ve **32 GB RAM** kaynakla tam performans gösterebilmelidir. En az **4 adet sanal ağ arayüzü** tanımlanabilmelidir.

2.6. Sistem, IPv4 ve IPv6 protokollerini tam olarak desteklemelidir.

2.7. SSL sertifikalarının yönetimi ve şifre çözme (offload) işlemleri, sistemin web tabanlı yönetim arayüzü veya REST API aracılığıyla yapılabilmelidir.

2.8. Sistem, reverse proxy mimarisinde çalışmalı ve tüm gelen-giden trafiği kontrol ederek yönlendirebilmelidir.

2.9. WAF çözümü, hem pasif (algılama) hem de aktif (engelleme) modlarda çalışabilmeli; bu modlar ayrı ayrı yapılandırılabilir olmalıdır.

2.10. WAF, OWASP Top 10 kapsamındaki güvenlik açıklarına karşı koruma sağlamalıdır (örneğin SQL Injection, Cross Site Scripting, CSRF vb.).

2.11. HTTP istek ve yanıt başlıklarının yeniden yazılmasını (rewriting) desteklemelidir.

2.12. Ürün; bilgi sızıntısını önlemek üzere http başlıklarının silinmesi/değiştirilmesi, özel hata sayfalarıyla detayların gizlenmesi, yanıtta ve loglarda duyarlı verinin maskelemesini desteklemelidir.

2.13. Yanıt gövdesi üzerinde analiz yaparak kredi kartı gibi hassas verilerin dışarıya sızmasını önleyebilmelidir.

2.14. Hem pozitif (izin verilen) hem de negatif (engellenen) güvenlik modeli desteklenmelidir.

2.15. Trafik bazlı hız sınırlamaları uygulanabilmeli, brute-force ve benzeri ataklara karşı dinamik koruma sunulmalıdır.

2.16. Olay ve trafik logları, syslog sunucularına veya üreticinin desteklediği merkezi log toplama sistemlerine aktarılabilmelidir.

2.17. WAF yönetimi LDAP, RADIUS gibi dizin servisleri ile entegre olabilmeli; Rol Tabanlı Erişim Kontrolü (RBAC) uygulanabilmelidir.

2.18. Sistem, Splunk, ArcSight gibi yaygın SIEM çözümleriyle entegrasyon sağlayabilmelidir.

2.19. HTTP/1.1 ve HTTP/2 protokollerini tam olarak desteklemelidir.

2.20. HTTPS desteği bulunmayan uygulamalar için, istemci ile WAF arasında HTTPS; WAF ile uygulama sunucuları arasında HTTP bağlantısı kurulabilmeli, otomatik sertifika dağıtımı yapılabilmelidir.

2.21. Sistem, Avrupa Birliği Genel Veri Koruma Tüzüğü (GDPR) ile uyumlu olmalı, kişisel verilerin korunmasını sağlayacak önlemleri desteklemelidir.

2.22. XML ve JSON formatındaki API trafiği için özel koruma modülleri içermelidir.
(Tercih sebebidir: Teklif edilen çözümün PCI-DSS standardına uyumlu olması ve bu durumun üretici beyanı veya geçerli sertifikalarla belgelenmesi.)

2.23. Sistem, uygulama seviyesinde L7 DoS, brute-force ve session hijack tespiti; istemci itibarı, parmak izi ve/veya davranış analizine dayalı yöntemlerle yapılabilecektir. Device ID/Fingerprint tek yöntem olarak zorunlu tutulmayacak; üretici mimarisiyle uyumlu eşdeğer yöntemler kabul edilecektir.Aynı şekilde analiz raporlarında da cihaz kimlik bilgisine göre filtreleme yapılabilmelidir.

3. YÜK DENGELEME HİZMETİ İÇİN GENEL ŞARTLAR

3.1. Aşağıdaki yük dengeleme algoritmalarını desteklemelidir:

- Round Robin
- Weighted Round Robin
- Least Connection
- Fastest Response

3.2. İçerik tabanlı yönlendirme (Content Switching) desteği bulunmalı; trafik, URL, HTTP Header, Cookie, Kaynak IP veya SSL Sertifikası SNI bilgilerine göre yönlendirilebilmelidir.

3.3. Trafik yönetimi hem Layer 4 hem de Layer 7 seviyelerinde yapılabilmelidir.

3.4. SSL bağlantıları uçta sonlandırılabilmeli; farklı SSL profilleri tanımlanabilmeli ve yönetilebilmelidir.

3.5. Oturum devamlılığı (Session Persistence) aşağıdaki yöntemlerle sağlanabilmelidir:

- Source IP

- Cookie
- SSL Session ID

3.6. HTTP trafiğini otomatik olarak HTTPS'ye yönlendirme (Redirect to HTTPS) desteği bulunmalıdır.

3.7. Trafik kaybı durumlarında otomatik sunucu yönlendirme (Failover) gerçekleştirilmeli; çözüm aktif/aktif veya aktif/pasif yüksek erişilebilirlik (HA) modlarında çalışabilmelidir.

3.8. Sağlık kontrolleri (Health Check) HTTP ve HTTPS üzerinden yapılmalı; ICMP ve TCP desteği tercih sebebidir.

3.9. HTTP trafiği için GZIP ve DEFLATE gibi veri sıkıştırma algoritmaları desteklenmelidir.

3.10. Sistem, REST API, webhook veya üreticiye özel izleme servisleriyle entegre çalışabilmeli; SNMP desteği varsa tercih sebebidir.

4. HİZMET ENTEGRASYON GEREKLİLİKLERİ

4.1. Sistem, istemci ile uygulama sunucuları arasında tüm trafiği yönlendirecek şekilde **reverse proxy mimarisinde** veya üreticinin önerdiği eşdeğer bir yapı üzerinde çalışmalıdır.

4.2. SSL/TLS trafiği uçta çözümlenmeli (offload) ve arka uç sistemlerle şifresiz veya yeniden şifrelenmiş (re-encrypted) iletişim kurulabilmelidir.

4.3. Şifreli HTTPS trafiği üzerinde uygulama katmanı seviyesinde analiz yapılabilmeli; aşağıdaki özelliklerden en az biri desteklenmelidir:

- Sertifika tabanlı kimlik doğrulama
- Sertifika içeriğinin uygulamaya aktarılması
- WAF ile uygulama sunucusu arasındaki trafiğin yeniden şifrelenmesi
- Seçilebilir cipher suite yapılandırması

4.4. WAF çözümü, HTTPS desteği bulunmayan uygulamalar için istemciyle **otomatik HTTPS bağlantısı kurabilmeli**, gerektiğinde geçici veya self-signed sertifika üretimi sağlayabilmelidir.

5. GÜVENLİK GEREKLİLİKLERİ

5.1. WAF çözümü, pasif (algılama) ve aktif (önleme) güvenlik modlarında çalışabilmeli; geçişler kontrollü ve test edilebilir şekilde yönetilebilmelidir.

5.2. Aşağıdaki saldırı türlerine karşı koruma sağlamalıdır (OWASP Top 10):

- SQL Injection (SQLi)
- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF)
- Command Injection
- Directory Traversal
- Insecure Deserialization

5.3. HTTP isteği ve yanıtı üzerinde aşağıdaki işlemler yapılabilmelidir:

- Header ekleme, değiştirme, silme
- URL yeniden yazma (rewriting)
- Yanıt içeriğinde hassas bilgi gizleme (ör. kredi kartı, kişisel veri)

5.4. Sunucu tipi ve sürüm bilgilerinin gizlenmesi; header silme/değiştirme, özel hata sayfaları ve duyarlı veri maskelemesi yöntemleriyle sağlanacaktır.

5.5. Pozitif ve negatif güvenlik modelleri (izin verilenleri kabul / yasaklıları engelle) birlikte desteklenmelidir.

5.6. Hız tabanlı saldırılar (rate-based attacks) ve brute-force girişimleri algılanmalı; atak kaynağı otomatik olarak engellenebilmelidir. Oturum bazlı anomali tespiti desteklenmelidir.

5.7. Form tampering'e karşı içeriğe dayalı bütünlük koruması sağlanmalıdır.

5.8. CSRF saldırılarına karşı token tabanlı koruma ve benzersiz URL üretimi desteklenmelidir.

5.9. Aşağıdaki gelişmiş güvenlik denetimleri uygulanabilmelidir:

- HTTP method kısıtlamaları
- Request body / header uzunluk sınırlandırmaları
- Cookie güvenliği ve replay attack önleme

5.10. Yüklenen içerikler üzerinde zararlı yazılım taraması yapılabilmeli veya bu işlem entegre bir çözümle sağlanabilmelidir.

5.11. WAF, uygulama seviyesinde DDoS saldırılarına karşı yerleşik koruma sunmalı; bu özellik için ek lisans gerekmemelidir.

5.12. XML ve JSON veri trafiği için özel koruma sağlanmalı; XML Schema doğrulama ve JSON Payload denetimi (boyut, nesne derinliği) yapılabilmelidir.

5.13. Sistem, zafiyet taraması sonucunda tespit edilen açıklardan, otomatik olarak giderilmesi durumunda uygulama ile ilgili sorun oluşturabileceklerini ayrı bir sayfada listeleyebilmeli ve bunların elle giderilmesi için ne yapılması gerektiği konusunda ek bilgi sağlayabilmelidir.

5.14. Sistem, realtime dinamik olarak policy oluşturabilmeli, otomatik self-learning ve policy oluşturma özelliğine sahip olmalıdır. Bu sayede zafiyetleri keşfetme ve hızlı kurulum

özelliklerine sahip olmalıdır. Çift yönlü çalışıp data ve protocol seviyesinde güvenlik sağlamalıdır.

5.15. Sistem, web uygulaması kapsamındaki, kullanıcı kimlik doğrulama sayfaları ve formlarını (login pages and forms) ek konfigürasyona gerek olmadan otomatik olarak tespit edebilmelidir. Tespit edilen sayfa ve formlar için şifre ataklarına karşı koruma mekanizmalarını (brute force attack protection) otomatik olarak devreye alabilmelidir.

5.16. Uygulama Güvenlik Duvarının, **tercihen** WebSocket protokolünü desteklemeli ve bu protokol üzerinden iletilen uygulama trafiğini analiz edebilmelidir. WebSocket trafiği özelinde aşağıdaki güvenlik özellikleri sağlanmalıdır:

- WebSocket bağlantılarını protokol seviyesinde parse edebilmeli ve oturumları analiz edebilmelidir.
- WebSocket uygulamaları için aşağıda listelenen atak türlerine karşı koruma sağlayabilmelidir:

    - Server stack abuse
    - Session riding
    - Cross-Site Request Forgery (CSRF)
    - Information leakage
    - Cross-Site Scripting (XSS)
    - SQL injection
    - Command shell injection
    - Server exploits
    - Cache poisoning
    - Buffer overflow
    - Exhausted server socket resources

  Bu korumalar; içerik analizi, imza tabanlı tespit, davranış analizi ya da üreticiye özel güvenlik teknikleri aracılığıyla sağlanabilmelidir.WebSocket mesajları için meta karakter, boş karakter (null byte), geçersiz yapı kontrolleri gibi temel protokol kontrolleri yapılabilmelidir.

5.17. Uygulama Güvenlik Duvarı, **tercihen** WebSocket tabanlı uygulamalarda uygulama katmanı güvenliği sağlayabilmeli ve aşağıdaki kontrolleri desteklemelidir:

- WebSocket bağlantılarında ws:// ve wss:// üzerinden gelen oturumları (login session) denetleyebilmeli,
- Belirli başlıkların (headers) zorunlu tutulmasına izin verebilmeli ve beyaz listede yer almayan kaynaklardan (origin) gelen bağlantıları engelleyebilmelidir.
- Gönderilen metin mesajlarının içeriği imza tabanlı (signature-based) tehdit tespiti ile analiz edilebilmeli, şüpheli durumlarda bağlantı kesilerek güvenlik kaydı oluşturulmalıdır.
- WebSocket protokolü kapsamında RFC uyumluluğu kontrol edilebilmeli; geçersiz karakterler, null byte ve özel karakter içeren mesajlar engellenebilmelidir.
- Uygulama belleğinin zehirlenmesini (cache poisoning) önlemek adına istemciden gelen metin mesajlarının maskelemeli olması şartı konulabilmelidir.

- Mesaj boyutu ve çerçeve (frame) boyutu sınırlanabilmeli; JSON içerikli mesajlar için yapı doğrulaması (payload validation) yapılabilmelidir.
- Mesaj gönderim süresi ve iki mesaj arası süre sınırlandırılabilmelidir.

5.18. Uygulama Güvenlik Duvarı, yaygın web uygulama tehditlerine karşı oluşturulmuş kapsamlı ve sürekli güncellenen saldırı imza setlerini (signature sets) içermelidir. Bu imzalar, hem genel güvenlik açıklarına (ör. CVE tabanlı) hem de uygulama profiline özel tehditlere karşı koruma sağlayacak şekilde üretici tarafından düzenli olarak güncellenmelidir.Ayrıca sistem, farklı uygulama türlerine veya güvenlik seviyelerine yönelik ön tanımlı politika şablonları (ör. temel koruma, sıkı koruma vb.) sunmalı ve bu şablonlar sayesinde hızlı kurulum ve yaygın kullanım senaryolarında etkin WAF yapılandırması yapılabilmelidir.

5.19. Uygulama Güvenlik Duvarı, güncel tehdit istihbaratına dayalı olarak otomatik saldırı imzası (attack signature) güncellemelerini desteklemeli, yeni ortaya çıkan zafiyet ve saldırı tekniklerine karşı sistemin proaktif olarak korunmasını sağlamalıdır. Bu imzalar, uygulama trafiğinde gerçek zamanlı analiz yaparak şüpheli davranışları tespit edebilmeli ve otomatik olarak savunma mekanizmalarını devreye almalıdır.

5.20. Uygulama Güvenlik Duvarı, Layer 7 (Application Layer) seviyesinde gerçekleşen DDoS saldırılarına karşı koruma sağlayabilmeli, **tercihen** makine öğrenimi (ML) tabanlı anomali tespiti ile normal trafik davranışını öğrenerek istisnai durumları ayırt edebilmeli, kullanıcı ve oturum bazlı davranış analizleri yapabilmeli ve dinamik rate limiting (oran sınırlama) mekanizmaları ile anlık saldırılara karşı esnek, otomatik yanıtlar üretebilmelidir.

5.21. Uygulama Güvenlik Duvarı, istemci trafiğini analiz ederek anormal veya şüpheli davranış sergileyen erişimleri tespit edebilmeli; kullanıcı ile bot trafiğini ayırt etmek amacıyla etkileşimli doğrulama yöntemlerini (ör. CAPTCHA, JavaScript challenge vb.) desteklemelidir. Bu doğrulamalar, dinamik risk değerlendirmesi, IP davranış analizi veya oturum bazlı anomaliler gibi tetikleyici koşullara dayalı olarak devreye alınabilmelidir. Bu yapı, kullanıcı deneyimini olumsuz etkilemeden özelleştirilebilir olmalıdır.

5.22. Uygulama Güvenlik Duvarı, **tercihen** sınırsız öğrenme moduna sahip olmalı, uygulama trafiğini sürekli analiz ederek normal davranış kalıplarını otomatik olarak öğrenebilmeli ve bu doğrultuda adaptif WAF kural güncellemeleri gerçekleştirebilmelidir. Öğrenme süreci, false positive (yanlış pozitif) oranlarını minimize edecek şekilde optimize edilmeli, gerçek kullanıcı trafiğiyle uyumlu hassasiyet seviyeleri belirlenerek güvenlik ile kullanıcı deneyimi arasında denge sağlanmalıdır.

5.23. Uygulama Güvenlik Duvarı, hedef uygulamanın altyapısında kullanılan sunucu (web server, application server) ve yazılım teknolojilerini (işletim sistemi, web sunucu yazılımı, programlama dili sürümü vb.) otomatik olarak tespit edebilmelidir ve tespit edilen teknoloji versiyonuna ait zafiyetleri bildirmelidir.

5.24. Uygulama Güvenlik Duvarı, gelişmiş scripting desteği sunmalı; trafik yönetimi, güvenlik politikalarının uygulanması ve özel senaryoların esnek şekilde gerçekleştirilmesine imkân tanımalıdır. Bu scripting yetenekleri, HTTP ve diğer protokol katmanlarında çalışarak dinamik koşullara göre trafik yönlendirme, modifikasyon veya engelleme işlemlerini desteklemelidir.

5.25. Uygulama Güvenlik Duvarı, **tercihen** oluşturulan WAF (Web Application Firewall) kurallarının OWASP Top 10 web uygulama güvenliği standartlarına uyumluluğunu analiz edebilmeli, mevcut yapı için OWASP Top 10 güvenlik skoru üretebilmeli ve bu skora dayalı olarak riskli alanları belirleyerek güvenlik sıkılaştırma (hardening) önerileri sunmalıdır.

6. YÖNETİMSEL GEREKLİLİKLER

6.1. YÖNETİM VE ERİŞİM DENETİMİ

6.1.1. WAF çözümünün yönetimi, HTTPS erişimli web tabanlı bir kontrol paneli veya REST API üzerinden yapılabilmelidir.

6.1.2. Sistem, Rol Tabanlı Erişim Kontrolü (RBAC) desteklemeli; kullanıcılar için farklı yetki seviyeleri (okuma, yazma, yönetici vb.) tanımlanabilmelidir.

6.1.3. LDAP ve RADIUS gibi dizin servisleriyle kimlik doğrulama entegrasyonu sağlanabilmelidir.

6.1.4. Yönetim erişimi için Çok Faktörlü Kimlik Doğrulama (MFA) desteği bulunmalıdır.

6.1.5. Ayrılmış görevler için farklı kullanıcı rolleri (örneğin operatör, uygulama yöneticisi, denetçi) yapılandırılabilmelidir.

6.1.6. Güvenilir IP adresleri (Trusted Hosts) tanımlanarak belirli kaynaklara özel yönetim erişimi sınırlandırılabilmelidir.

6.1.7. Yanıt sayfalarının içeriği özelleştirilebilmelidir (ör. hata mesajları, engelleme sayfaları).

6.2. GÜNLÜKLEME, OTOMASYON VE RAPORLAMA

6.2.1. WAF, saldırı kayıtları, trafik hareketleri ve yönetim işlemlerini detaylı olarak loglayabilmeli; loglar syslog, CEF, LEEF gibi yaygın formatlarda dış sistemlere (SIEM, log sunucusu) aktarılabilmelidir. Gerekirse loglar, harici log analiz platformlarında Common, W3C Extended ve NCSA Extended gibi standart web log formatlarına dönüştürülebilmelidir.

6.2.2. Loglar, syslog/CEF/LEEF gibi standart protokoller ile log toplayıcılara aktarılacak; Elasticsearch/ Azure Event Hub gibi hedeflerle entegrasyon Logstash/Beats/agent veya üreticinin desteklediği aracı bileşenler üzerinden sağlanmalıdır.

6.2.3. Hassas veriler loglarda maskelemeye (suppress) tabi tutulabilmelidir.

6.2.4. Her saldırı kaydı için benzersiz bir log ID üretilebilmelidir.

6.2.5. SIEM çözümleri (ör. Splunk, ArcSight) ile tam entegre çalışabilmelidir.

6.2.6. WAF, güvenlik politikalarında oluşan false positive durumlarını analiz raporlarında gösterebilmeli; öğrenme (learning) modunda, yöneticinin onayıyla güvenlik politikalarının iyileştirilmesine imkân tanımalıdır.

6.2.7. Farklı uygulamalar için özel güvenlik politikaları oluşturulabilmeli, regex desteği ile özel imzalar tanımlanabilmelidir.

6.2.8. WAF yapılandırma dosyaları için otomatik ve zamanlanmış yedekleme yapılabilmelidir.

6.2.9. PCI-DSS uyumluluğu kapsamında özel raporlar ve atak tespiti sağlanabilmelidir.

6.2.10. Trafik, saldırı ve sistem performans raporları zamanlanarak dış sistemlere gönderilebilmelidir (e-posta, API, SFTP vb.).

6.2.11. WAF, güvenlik politikalarında oluşan false positive durumlarını analiz raporlarında gösterebilmeli; öğrenme (learning) modunda, yöneticinin onayıyla güvenlik politikalarının iyileştirilmesine imkân tanımalıdır.

6.2.12. SNMP trap veya eşdeğer API tabanlı alarm mekanizmaları desteklenmelidir.

6.2.13. Sistem, Chef, Puppet, Ansible gibi otomasyon araçlarıyla entegrasyon sağlayacak JSON tabanlı REST API desteğine sahip olmalıdır.

7. XML GÜVENLİK DUVARI

7.1. WAF çözümü, XML tabanlı Web Hizmetleri için yaygın protokolleri desteklemeli ve XML'e özgü saldırılara karşı koruma sağlamalıdır.

7.2. Sistem, **WS-I Basic Profile** ile uyumlu SOAP servislerini desteklemeli ve bu profile uygun şekilde yapılandırılabilir güvenlik kuralları sunmalıdır.

7.3. XML dokümanlarının yapısal doğrulaması desteklenmeli; bu sayede aşağıdaki saldırı türlerine karşı koruma sağlanmalıdır:

- XML tabanlı DoS/DDoS saldırıları
- XML Injection (SQL, OS veya XSS tabanlı)

7.4. SOAP mesajları, başlık ve gövde düzeyinde analiz edilebilmeli; **tercihen** şema doğrulama (schema validation) desteği bulunmalıdır.WSDL desteği mevcut ise otomatik şema çıkarımı yapılabilmesi tercih sebebidir.

7.5. Uygulama Güvenlik Duvarı, XML formatlı trafiği doğrulayabilmeli, XML şema ve WSDL dokümanlarını temel alarak uyumluluk kontrolü gerçekleştirebilmelidir. XML içeriğinde hassas

verilerin maskelemesi yapılabilmeli, SOAP mesajları üzerinde tanımlanmış bölümler içinşifreleme veya imza kontrolleri (gerekirse uygulama sunucusu veya API Gateway desteğiyle) gerçekleştirilebilmelidir.

8. API KORUMA (JSON)

8.1. WAF çözümü, JSON tabanlı API altyapıları ve web servisleri için yaygın API tehditlerine ve JSON'a özgü saldırı türlerine karşı koruma sağlamalıdır.

8.2. Sistem, JSON payload doğrulamasını desteklemeli; en az aşağıdaki kontrolleri uygulayabilmelidir:

- Veri boyutu ve içerik limiti denetimi
- JSON anahtar uzunluğu kontrolü
- Maksimum nesne derinliği (object depth) sınırlandırması

8.3. OpenAPI (Swagger) spesifikasyon dosyaları içe aktarılabilmeli ve sistem bu tanımlar üzerinden otomatik API güvenlik kuralları oluşturabilmelidir.

8.4. API taleplerine yönelik **rate limiting** (hız sınırlama) mekanizmaları uygulanabilmelidir.

8.5. API istek metodları (GET, POST, PUT, DELETE vb.) filtrelenebilmeli ve yetkilendirme politikalarına göre sınırlandırılabilmelidir.

8.6. Uygulama Güvenlik Duvarının REST API desteği olmalıdır ve bu sayede sisteme bağlanmadan, program ve kod ile dışarıdan güvenlik politikası detayını görme ve güvenlik politikası üzerinde değişiklik yapmaya imkân verebilmelidir.

8.7. API güvenlik politikaları, pozitif güvenlik modeli prensibine uygun olarak yapılandırılmalı; sadece tanımlı ve izin verilen API isteklerinin işlenmesine izin verilmelidir.

8.8. Uygulama Güvenlik Duvarı API isteklerini inceleyerek atakları tespit etme yeteneğine sahip olmalıdır.

9. UYGULAMA TRAFİĞİ YÖNLENDİRME VE YÜK DENGELEME YETENEKLERİ

9.1. WAF çözümü, Load Balancer (Yük Dengeleyici) işlevi ile uygulama trafiğini optimize edecek şekilde çalışabilmelidir. Trafik yönlendirme algoritmaları, 3.2 maddesinde belirtilen yöntemleri destekleyecek şekilde yapılandırılmalıdır.

9.2. Sistem, arka uç sunucuların sağlık durumunu HTTP, HTTPS veya TCP üzerinden izleyebilmeli; arıza tespiti durumunda trafiği otomatik olarak aktif sunuculara yönlendirebilmelidir.

9.3. Sistem, trafik yönlendirme işlemleri aşağıdaki kriterlere göre yapılabilmelidir:

- Kaynak IP adresi
- Cookie bilgisi
- HTTP header içeriği
- URL path bilgisi
- SSL Sertifikası SNI alanı

9.4. Sistem, içerik tabanlı anahtarlama (Content Switching) desteği sunmalı; farklı içerikler farklı sunucu gruplarına yönlendirilebilmelidir.

9.5. Sistem, gelen yanıtlar isteğe bağlı olarak önbelleğe alınabilmeli (response caching); sık erişilen veriler doğrudan WAF üzerinden sunulabilmelidir.

9.6. Sistem , giden yanıtlar için içerik türüne bağlı olarak veri sıkıştırma (compression) işlemleri uygulanabilmelidir. En az aşağıdaki içerik türleri desteklenmelidir:

- JavaScript
- XML
- JSON

11. RAPORLAMA, İZLEME VE PERFORMANS TAKİBİ

11.1. WAF çözümü, erişim trafiği, güvenlik olayları, saldırılar, sistem hataları ve yönetimsel işlemler için kapsamlı log kayıtları tutabilmelidir.

11.2. Sistem, en az aşağıdaki yaygın log formatlarını desteklemelidir:

- Common Log Format
- W3C Extended Log Format
- NCSA Extended

11.3. Log kayıtlarının dış sistemlere aktarımı için syslog ve güvenli dosya aktarımı yöntemleri veya üreticinin sunduğu log yayınarayüzleri kullanılmalıdır. Elasticsearch/Kibana gibi platformlar için connector üzerinden entegrasyon yapılması gerekmektedir.

11.4. Belirli güvenlik olayları için log kaydı devre dışı bırakılabilmeli; hassas veriler loglarda maskelemeye tabi tutulabilmelidir.

11.5. Olay ve trafik raporları zamanlanarak üretilebilmeli ve şu yollarla dış sistemlere gönderilebilmelidir:

- API
- E-posta
- Webhook
- S3 Bucket

11.6. Entegrasyon hedefleri (Azure Event Hub, Elasticsearch, Kibana vb.) aracı bileşenler (agent, Logstash, connector) üzerinden desteklenecek; Kibana görselleştirme katmanı olarak, verinin Elasticsearch'e yazılması esas alınmalıdır.

11.7. Yönetim yazılımı teklife dahil edilmelidir. Yönetim yazılımı en az aşağıdaki özellikleri desteklemelidir.

- En az Uygulama, Altyapı, Uygulama Güvenliği, Erişim noktası başlıklarında detayları içeren olay takip ve görüntüleme ekranına (dashboard) sahip olmalıdır. Bu ekran yönetici isteklerine göre düzenlenebilmelidir
- Cihazlar üzerinde bulunan genel uygulamalar, mikroservis uygulamaları, 3 katmanlı uygulamalar için servis grafiği/topolojisi gösterebilmelidir. İstemci-Cihaz-Sunucu hattında trafik durumunu görsel olarak grafik halinde sunabilmelidir. Bu sayede yönetici sorunları hızlıca tespit edebilmelidir
- WEB Uygulama güvenlik duvarı ve cihaz güvenliği incelenerek açık kapalı fonksiyonlar, şifreleme, loglama, imza konfigürasyonu, yönetim erişim güvenliği bilgilerini kullanarak güvenlik seviyesini (Safety index) ölçmelidir. Güvenlik seviyesinin artırılması amacı ile yapılabilecek iyileştirmeler hakkında önerilerde bulunabilmelidir.

12. YÜKSEK ERİŞİLEBİLİRLİK VE YÖNLENDİRME DAYANIKLILIĞI

12.1. WAF çözümü, sistem sürekliliğini sağlamak amacıyla aktif/aktif veya aktif/pasif mimarilerde **Yüksek Erişilebilirlik (HA)** desteği sunmalıdır.

12.2. Sistem, bağlantı yedekliliği sağlayabilmeli; arıza durumlarında trafik otomatik olarak çalışır durumdaki diğer bileşenlere yönlendirilebilmelidir (**failover** desteği).

12.3. Yapılandırma dosyaları, belirlenen zaman aralıklarında **otomatik ve zamanlanmış şekilde yedeklenebilir** olmalıdır.

12.4. WAF hizmeti, üreticinin sunduğu sanal ağ segmentasyonu (virtual domains, logical segmentation vb.) veya coğrafi mantıksal ayrım yapılarına entegre çalışabilecek nitelikte olmalıdır.

13. UYUMLULUK VE SERTİFİKASYON GEREKLİLİKLERİ

13.1. WAF çözümü, aşağıdaki güvenlik ve veri koruma standartlarıyla **uyumlu** olmalıdır:

- **PCI DSS (Payment Card Industry Data Security Standard)**
- **ISO/IEC 27001 (Bilgi Güvenliği Yönetim Sistemi)**
- **GDPR (General Data Protection Regulation)**

13.2. Teklif edilen ürün, üretici tarafından sağlanan aşağıdaki gibi **resmi güvenlik sertifikalarına** sahip olmalıdır:

- Common Criteria (EAL)
- FIPS 140-2
- veya eşdeğer güvenlik sertifikaları

13.3. WAF sistemi, **PCI DSS uyumluluk raporları** oluşturabilmeli ve **PCI atak tespiti** yapabilmelidir.

14. DESTEK, BAKIM VE EĞİTİM HİZMETLERİ

14.1. Teklif edilecek hizmet için en az 3 yıl boyunca üretici destek hizmeti sağlanmalıdır.

14.2. 7/24 erişilebilir üretici destek hattı sunulmalıdır.

14.3. Eğitim kapsamında, 3 personel için uygulamalı yönetim ve kullanım eğitimi verilmelidir. Eğitimlerde üretici firmanın standart sertifikalı eğitim materyalleri kullanılmalı ve eğitim sonunda katılımcılara sertifika sağlanmalıdır.

14.4. Eğitim uygulamalı olmalı, örnek senaryolarla desteklenmeli ve eksiksiz olmalıdır.

14.5 Eğitim, hizmet alındıktan sonra hemen ya da en geç 1 ay içinde verilmelidir.