TOBB
EKONOMİ VE TEKNOLOJİ
ÜNİVERSİTESİ

Bil527 Ağ Savunma Sistemleri

Saldırı Tespit Sistemleri(STS) Karşılaştırılması ve Suricata Kurulumu

Alperen Şahin 151117007

24.07.2016

Ankara

İçindekiler

Giriş

Hayatımıza akademik amaçlı bir araştırma ağı olarak giren internet, günümüzde önemli toplumsal dönüşümlere altyapı sağlar duruma gelmiştir. O zamanlar internetin bu kadar kapsamlı ve etkili kullanılabileceği öngörülemediğinden ya da önemsiz bir konu olarak nitelendirildiğinden olsa gerek internet ortamındaki güvenlik pek önemsenmemiş ve bu konuda yeteri kadar çalışma yapılmamış. Fakat internet kullanım oranının artması, internete bağlı kurum sayısının artması, internet ortamında yapılabilen işlerin çeşitliliğinin artması neticesinde güvenlik konusu ister istemez ciddi bir problem haline gelmiştir. Özellikle 1988 yılında ortaya çıkan Morris solucanının[1], başarılı bir şekilde binlerce bilgisayar sistemine sızmayı başarması ve sızdığı bilgisayar sistemlerini çalışamaz hale getirmesi büyük bir faciaya neden olmuş ve bu olaydan sonra internet ortamındaki güvenlik konusunda farkındalık oluşmaya başlamıştır. Bu olaydan sonra bilgi güvenliği konusunda çalışmalar hız kazanmış ve 90'lı yılların başlarında ilk güvenlik duvarı uygulamaları ile bir takım teknik güvenlik önlemlerinin alınması konusunda referans çalışmalar başlamıştır.

Güvenlikle ilgili tehditlerin sayısının ve türlerinin hızla artmasına karşılık geliştirilen güvenlik önlemlerinde de hızlı bir gelişim yaşanmaktadır. Bu kapsamda bilgisayarların güvenliğini sağlamak, yetkili olmayan kişilerin sistemlere erişerek bilgileri ele geçirmelerini veya değiştirmelerini engellemek için güvenliğin ilk basamağı olarak kimlik doğrulama ve erişim kontrolü gibi güvenlik mekanizmaları geliştirilmiştir. Fakat internet ve iletişimin artmasıyla beraber kötü niyetli kullanıcılar tarafından saldırılıp zarar verilebilecek daha çok sistem ve elde edilebilecek daha çok bilgi ortaya çıkmaya başlamış ve buna bağlı olarak gerçekleştirilen saldırı sayısında ve kullanılan saldırı yöntemlerinde de ciddi artışlar gözlemlenmiştir. Örneğin bir yer sağlayıcı firmasının paylaştığı rapora göre sadece o hosting firmasına karşı yapılan saldırılardan dolayı saldırı tespit sistemleri tarafından bir iş gün içerisinde 190 milyon adet IDS alarmı üretilmektedir.

SALDIRI TESPİT SİSTEMLERİ

Nedir?

Genel olarak yapılan saldırıların büyük bir çoğunluğu kullanılan sistemlerin zaafları ve/veya açıklıklarından faydalanılarak gerçekleştirilmektedir. Bu tür saldırıları engellemenin iki türlü yöntemi vardır. Birincisi tamamen güvenli bir sistem ve ortam oluşturmak, ikincisi ise en kısa zamanda saldırıların tespit edilip gerekli önlemlerin alınmasının sağlanmasıdır. İlk yöntem bu güne kadar pek mümkün olmadı ve olası da görünmüyor. Onun için bir sistemin güvenliği, sistem güvenlik sorumluları tarafından rutin kontrolleri yapılmak kaydı ile saldırı gelene kadar bekleme pozisyonunda kalarak, saldırı geldiğinde olabildiğince hızlı bir şekilde saldırıyı tespit edip gerekli önlemi alabilmeyi mümkün kılacak şekilde tasarlanmalıdır. İşte bu aşamada da devreye saldırı tespit sistemleri girmektedir. En genel anlamıyla, saldırı tespiti işini yapmak için geliştirilen sistemlere "saldırı tespit sistemleri" denilmektedir. 1980 yılında James Anderson'ın yaptığı tanımdan[2] günümüze kadar yapılan araştırmalar ve çalışmalar neticesinde saldırı tespit sistemleri için farklı tanımlar yapılmıştır. Bu tanımlar yanlış olmamakla birlikte sadece günümüzdeki saldırı tespit sistemleri tanımının yanında biraz eksik kalmaktadır. Örneğin yapılan tanımlardan bazıları şöyledir:

- Bilgisayar sistemlerine yapılan atakları ve kötüye kullanımları belirlemek için tasarlanmış sistemlerdir,

- Tercihen gerçek zamanlı olarak, bilgisayar sistemlerinin yetkisiz ve kötüye kullanımı ve suiistimalini tespit etmek için kullanılırlar,
- Kullanım alanı ve türüne bağlı olarak saldırıyı engelleyebilen veya saldırıyı durdurma girişiminde bulunmayan, olası güvenlik ihlali durumlarında sistem güvenlik çalışanlarına uyarı mesajı veren sistemlerdir,
- Bilgisayar sistemlerinin kaynaklarına veya verilerine yetkisiz erişimleri tespit edebilen sistemlerdir,
- Bilgisayar ortamındaki "hırsız alarm"larıdır.

Günümüzde kullanılan tanımı ise tüm bu yapılan tanımları kapsamaktadır. Saldırı tespit sistemleri, bilginin elektronik ortamlarda taşınırken, işlenirken veya depolanırken başına gelebilecek tehlike ve tehditlerin ortadan kaldırılması veya bunlara karşı tedbir alınması amacıyla, bilgiye yetkisiz erişim ve bilginin kötüye kullanılması gibi internet veya yerel ağdan gelebilecek çeşitli paket ve verilerden oluşan girişimleri tespit edebilme, bu tespitleri sms, e-posta veya SNMP mesajları ile sistem güvenliğinden sorumlu kişilere iletebilme ve gerektiğinde paketi/erişimi düşürebilme özelliğine sahip yazılımsal ve/veya donanımsal güvenlik araçları olarak tanımlanabilir.

Saldırı Tespit Sistemleri, internet dünyasının gelişim sürecinde özellikle tüm dünyada kullanılan web trafiğinin artması ve de web sayfalarının popüler hale gelmesi ile birlikte kişisel ya da tüzel sayfalara yapılan saldırılar sonucu ihtiyaç duyulan en önemli konulardan biri haline gelmiştir. Bununla birlikte kurum ya da kuruluşların sahip oldukları ve tüm dünyaya açık tuttukları Mail, DNS ve Web gibi sunucularının benzeri saldırılara maruz kalabilecekleri ihtimali yine saldırı tespit sistemlerini internet güvenliği alanının vazgeçilmez bir parçası haline getirmiştir. Yine kurumların sahip oldukları çalışanların kendi kurumlarındaki kritik değer taşıyan yapılara/verilere saldırabilme/zarar verme ihtimalleri düşünülünce iç ağın ya da tek tek kritik sunucuların kontrol altında tutulma gerekliliği de saldırı tespit sistemlerinin kullanımını kaçınılmaz kılmıştır.

Bilgi güvenliği alanında 4 kitap ve 140'tan fazla da makale yazmış olan başarılı araştırmacı Denning, saldırı tespit sistemlerinin gerekliliği konusunda yaptığı çalışmalar neticesinde 1986 yılında yayınlamış olduğu makalesinde bu durumları özetler nitelikte şunları söylemektedir[3]:

> *Mevcut sistemlerin çoğunda saldırıya, sızmaya ve muhtelif diğer biçimlerde zarar verilmesine imkân verecek zaaflar bulunmaktadır; tüm bu zaafların bulunması ve düzeltilmesi teknik ve/veya ekonomik nedenlerden ötürü mümkün olamamaktadı., Bilindik zaafları olan mevcut sistemler, daha yüksek güvenlik sağlayan alternatifleri ile eğiştirilememektedir. Bunun ana nedeni ya mevcut sistemlerde var olan bazı özelliklerin daha yüksek güvenlik sağlayan alternatiflerinde var olmaması ya da ekonomik nedenlerle değiştirilememesidir. Mutlak güvenliğe sahip sistemlerin geliştirilmesi imkânsız değilse bile son derece güçtür.En yüksek güvenlik düzeyine sahip sistemler bile yetkilerini kötüye kullanan kullanıcıların zarar verebilmesine imkân tanır durumdadır.*

Artı ve Eksileri

Saldırı tespit sistemlerinin ***avantajları*** olarak şunlar söylenebilir:

Erken Tespit: Saldırı tespit sistemleri, gerçekleşen bir saldırıyı sistem güvenlik sorumlularından çok daha önce tespit ederek saldırı ile ilintili olarak sms veya e-posta gibi farklı

yollarla sorumlu kişileri anında uyarabilir ve oluşabilecek zararın etkisinin minimize edilmesine katkıda bulunurlar.

Detaylı Bilgi Toplanması: Saldırı tespit sistemleri sayesinde devam etmekte olan veya geçmişte gerçekleştirilmiş olan saldırılarla ilgili, saldırının kaynağı, büyüklüğü ve hedeflerinin saptanması noktasında son derece değerli bilgiler elde edilebilir.

Kanıt Niteliği: Saldırı tespit sistemleri tarafından toplanan bilgiler hukuki yollara başvurulduğunda kanıt olarak kullanılabilir.

Saldırı tespit sistemlerinin ***zayıflıkları*** olarak ise şunlar söylenebilir:

Paket Parçalama ve Zamanlama Saldırıları: Saldırı tespit sistemleri, paketleri analiz etmek için parçalanmış paketleri tekrar birleştirmek zorundadır. Uzun zaman aralıkları ile küçük parçalar halinde gönderilen paketler trafiğin aksamaması için bu sistemler tarafından tam olarak analiz edilememektedir.

Tarama Sırasının Karıştırılması: Sıra ile IP adreslerine veya portlara gerçekleştirilecek bir tarama, saldırı tespit sistemleri tarafından hemen tespit edilir. Fakat bu taramalar rastgele sırada yapılırsa analizi zorlaşabilir ve hatta bu yöntemle saldırı tespit sistemleri atlatılabilir.

Paket Kaçırma (False Positive, False Negative): Genel olarak saldırı tespit sistemlerinin hataları uyarılar vermesi sonucu paket kaçırması olarak görülebilir. Zararlı olmayan normal bir davranış için uyarı vermesi false pozitif; şüpheli olan bir davranış için uyarı vermeyerek paketin sorunsuz geçişine izin vermesi false negative'dir.

Kullanım Tipleri

Saldırı tespit sistemleri genel olarak Sunucu Tabanlı ve Ağ Tabanlı olmak üzere iki farklı tipte olmaktadırlar. Sunucu tabanlı saldırı tespit sistemlerinin görevi; kurulu bulunduğu sunucunun trafiğini, kayıt dosyalarını ve işlemlerini sunucu üzerinde bulunan ve o sunucuya göre özelleştirilmiş olan atak/imza veritabanı temel alınarak dinlemek ve atakları sezerek cevap vermektir. Ağ tabanlı saldırı tespit sistemlerin görevi ise ağ kartının geçirgen (promiscuous) moda getirilmesi ile ağ ya da ağlara yönlenmiş olan tüm trafiğin dinlenmesi, bu ağdan geçen her bir veri paketi içeriğinin sorgulanarak mevcut imzalarla karşılaştırılıp bir atak olup olmadığına karar vererek kaydını alabilmek, gerektiğinde atakları kesmek, sistem yöneticisini bilgilendirmek ve ilgili raporları oluşturabilmektir.

Çalışma Mantığı

Saldırı tespit sistemleri içerik olarak bilgi/öğrenme tabanlı (anormallik tespiti) ve imza(kötüye kullanım tespiti) tabanlı olmak üzere iki farklı mantığa göre çalışmaktadırlar. İlk yapıda sistemlerin ve ağın işleyişi belirli bir düzenle özdeşleştirilerek tanımlı ağ veya kullanıcı için eşik değerleri tanımlanır. Daha sonra takip edilen trafik bu eşik değerlerine göre değerlendirilerek, oluşacak herhangi bir normal dışı hareket ile saldırının tanımlanması hedeflenir. Yani bilgi/öğrenme tabanlı saldırı tespitinde ise, sistem kullanıcılarının, normal davranışlarından farklı olarak gösterdikleri davranış şekillerine göre çalışma yapılır. Bu yöntem, tahmine dayalı bir sistemdir ve genellikle "uzman sistemler" ve "bulanık mantık" teknolojilerinden faydalanılır. Bir saldırı tespit sisteminin, ağ üzerindeki faaliyetleri izlemek için ağ üzerindeki farklı noktalarda alıcı cihazlarını ve yazılımlarını kurmak gerekebilir. Bu cihaz ve yazılımların görevi, sorumlu oldukları ağ bölümü üzerinde gerçekleşen faaliyet bilgilerini, saldırı tespit sistemi merkezine aktarmaktır. Örneğin, web sunucusuna gelen

isteklerin %99'u "index.html" dosyasını çağırıyor ise cmd.exe dosyasını çağıran bir istek geldiğinde bu hemen fark edilecek ve bunun için uyarı mesajı üretilecektir. Çok daha mantıklı bir çalışma prensibi olmasına karşın bu tür sistemlerin normal olarak nitelendirilebilecek hareketleri öğrenmeleri oldukça fazla zaman almaktadır. Bundan dolayı bu hareketlerin zaman içerisinde değişebilirliği, kurulduğu sistemlerin yeniden yapılandırılması veya ağa yeni sistemlerin eklenmesi işleri daha da zorlaştırmakta ve saldırı tespit sistemlerinin paket kaçırma olasılığını daha da arttırmaktadır. İkinci yani imza tabanlı yapıda ise anti virüs sistemlerinde olduğu gibi oluşturulmuş çeşitli imzalar ile paketler incelenir ve saldırıların bu şekilde saptanması hedeflenir. Daha önce karşılaşılan saldırı şekilleri ayrıntılı olarak analiz edilerek elde edilen bilgiler, yani saldırının imzası, saldırı tespit sisteminin bilgi tabanına kaydedilir. Her tanımlanmış saldırının bir imzası vardır. Saldırı imzaları dışında kalan her faaliyet, normal olarak algılanır. Bu şekilde çalışan bir saldırı tespit sisteminin verimli çalışması için, sürekli saldırı imzalarını güncelleyerek sistemi, yeni saldırı tiplerini de tanıyıp tespit edebilecek şekilde güncel tutmak gerekir.

STS YAZILIMLARI

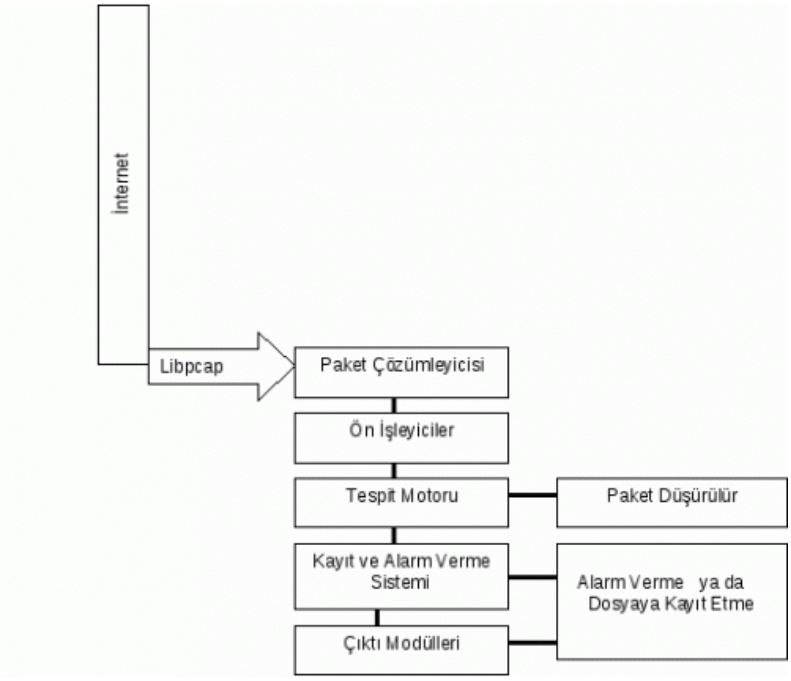
SNORT

GNU lisansı ile dağıtılan, açık kaynak kodlu ve ücretsiz bir yazılım olan Snort, 1988 yılında Martin Roesch tarafından geliştirilmiştir. Şu anda da Martin Roesch'un kurmuş olduğu Sourcefire firması tarafından geliştirilmesine devam edilen, dünya genelinde en çok kullanılan, Linux, Windows, MAC ve FreeBSD gibi birçok farklı platformda sorunsuz olarak çalışabilen, IP ağları üzerinde gerçek zamanlı trafik analizi ve paket loglaması yapabilen bir saldırı tespit ve önleme sistemi yazılımıdır. Genel olarak imza tabanlı olarak çalışan Snort, protokol ve anomali analizi yapabilme yeteneğine de sahiptir. Kullanıcıların kendi kurallarını yazabilmesine imkân sağlayan esnek bir kural diline sahip olmasının yanında snort.org ve emergingtreats.com adreslerinden indirilebilen ücretli veya ücretsiz kural setleri kullanılarak; yazılım protokol analizi, içerik tarama/eşleme, arabellek taşması, port taraması, CGI saldırısı, işletim sistemi parmak izi denemesi gibi pek çok saldırı ve zararlı/şüpheli yazılım çeşitlerini tespit edebilmektedir.

Snort'un Saldırı Tespit Sistemi (STS) olarak kullanıldığı durumlarda genellikle iki ağ arayüz kartı kullanılır. Bu arayüzlerinden birisi ağı dinlemek için, diğeri ise Snort'a uzaktan erişip Snort'un yapılandırılmasında kullanılır. Ağı dinleyen arayüze genellikle IP adresi atanmaz ve bağlı olduğu anahtarın(switch) tüm portları bu arayüze aynalanır (mirroring). Bu yöntemle, anahtar üzerinden geçen tüm paketlerin Snort tarafından dinlenilmesi sağlanmış olur.

Yapısı ve Özellikleri

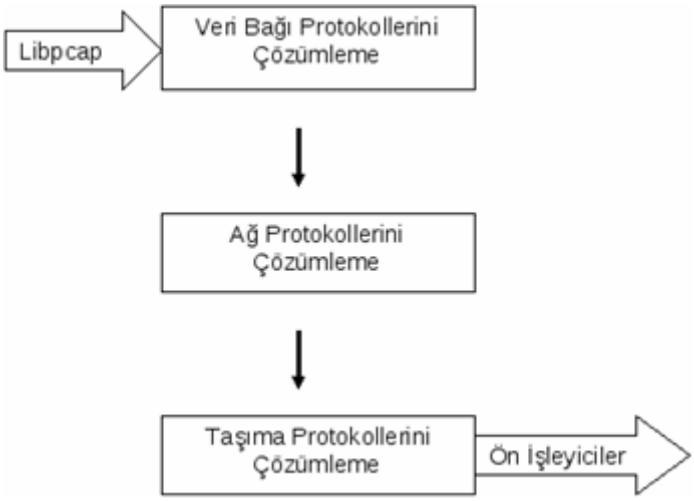
Snort'un mimarisi performans, basitlik ve esnekliğe dayalıdır. Paket çözücü, ön işleyici, tespit motoru ve günlükleme/alarm olmak üzere 4 temel bileşen üzerine inşa edilmiştir.



Şekil 1 – Snort Mimarisi

Libpcap (Packet Capture Library): Snort'un ağ kartından paketleri çekmek için Linux/Unix sistemlerde libpcap, Windows sistemlerde ise WinPcap olarak kullandığı paket yakalama kütüphanesidir.

Paket Çözümleyici (Decoder): Paket yakalama kütüphanesinin yakalayıp gönderdiği veri bağı yani 2. katman verisini alır ve ayrıştırarak (2.katman için Ethernet/802.11, 3. katman için IP/ICMP , 4. katman için tcp/udp gibi) sonraki aşamalarda işlenmek üzere ya da direk tespit motoruna gönderilmek üzere hazır hale getirir.



Şekil 2 – Snort Paket Çözümleyici Veri Akışı

Ön işleyici (preprocessor): Yakalanan bir paketin tespit motorunda gerçekleştirilecek olan kural uygulamaları öncesinde işlenmeye hazır hale getirilmesi gereklidir. Örneğin, paket parçalanmış bir yapıda ise paketin boyutunun tespitinden önce tüm parçaların yeniden bir araya getirilmesi gereklidir. İşte ön işleyiciler, paket çözümleyicisi tarafından çözümlenmiş olan paketlerin Snort tarafından daha kolay kavranabilmesi ve anlaşılabilmesi adına daha anlamlı parçalar haline getirir. Snort yapılandırma dosyasından aktif edilebilir ya da devre dışı bırakılabilir bir yapıdadır. Örneğin, port tarama ön işleyicisi aktif hale getirilirse, sistem üzerinde yapılacak olan herhangi bir port tarama işlemi Snort tarafından başarı ile yakalayacaktır.

Tespit Motoru (Detection Engine): Tespit motoru bileşeni Snort'un en önemli kısmı, kalbi olarakta nitelendirilmektedir. Tespit motorunun görevi, paket çözümleyicisi ve ön işleyici bileşenlerinden gelen paketlerde saldırı faaliyeti mevcut ise tespit etmektir. Bu amaçla tespit motoru Snort kurallarını kullanmaktadır. Snort, tüm kuralları başlangıçta okur ve ağaç düğüm yapısını ağdan toplanan paketlere uygulamak üzere oluşturur. Eğer bir paket herhangi bir kural ile eşleşirse, uygun eylem gerçekleştirilir aksi takdirde paket düşürülür. Uygun eylem, paketin kaydedilmesi ya da alarm verme olabilmektedir.

Kayıt ve Alarm Verme Sistemi, Çıktı Modülleri: Tespit motoru, ağ içinde akan paketler içerisinde yapmış olduğu tespitlere bağlı olarak saldırı olarak öngördüğü paketler için uyarı mesajları

üretir ve bunlarla ilgili olarak da basit bir metin dosyasında, tcpdump formatında veya diğer kaydetme formatlarında log tutabilir. İşte çıktı modülleri de bu uyarıların nasıl olacağı ve nereye ne biçimde kaydedileceği konusunu yönetirler.

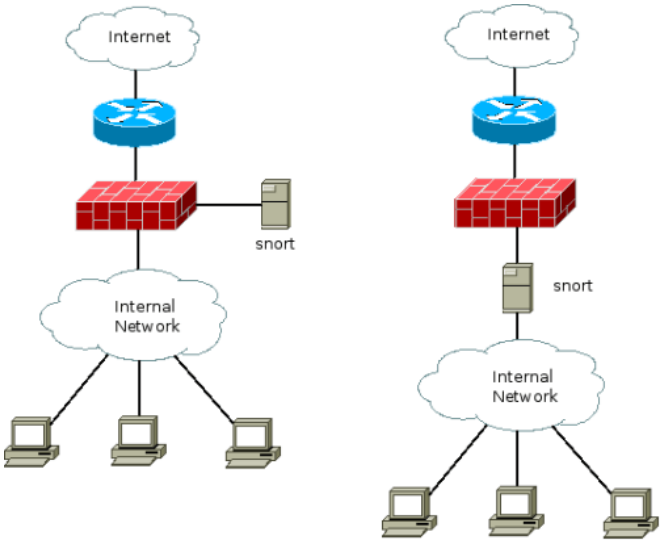
Çalışma Modları

Snort temel olarak paket izleme (packet sniffer), paket günlükleme (packet logger) ve sızma tespit/engellenme (IDS/IPS) olmak üzere üç farklı moda çalışabilecek şekilde yapılandırılabilmektedir.

Paket İzleyici Modu (packet sniffer): Snort'un sadece geçen paketleri izlemesi isteniyorsa bu modda çalıştırılır. Bu mod tcpdump paket izleyici programı gibi basit bir şekilde ağdan paketleri okuyup sürekli bir şekilde konsolda göstermektedir.

Paket Günlükleme Modu (packet logger): Snort, belirtilen parametrelere göre paketlerin istenilen formatta diske yazılması istendiğinde bu modda çalıştırılır.

Ağ Sızma Tespit/Engelleme Sistemi (NIDS/NIPS) Modu: Snort'un genel olarak kullanıldığı sızma girişimlerini tespit etme modudur. Snort bu modda temel olarak trafiği analiz ederek kullanıcı tarafından daha önceden tanımlanmış olan kurallarla karşılaştırma yaparak ilgili kurallarda belirtilmiş olan eylemlerin uygulanmasını sağlar.



Şekil 3 – Snort'un Ağ Topolojisinde Konumlandırılması

Kural Yazımı

Snort kuralları, kural başlığı ve kural seçenekleri olmak üzere mantıksal olarak iki kısma ayrılmaktadır:

1. Kural başlığı
   a. Kural eylemi,
   b. Protokol,
   c. Kaynak IP adresi,
   d. Hedef IP adresi,
   e. Alt ağ maskesi,
   f. Kaynak ve Hedef port bilgileri.
2. Kural seçenekleri:
   a. Uyarı mesajları ve paketin hangi bölümünün inceleneceğini bilgisini içerir.

Aşağıda örnek olarak Snort için yazılmış bir alarm kuralı gösterilmiştir:

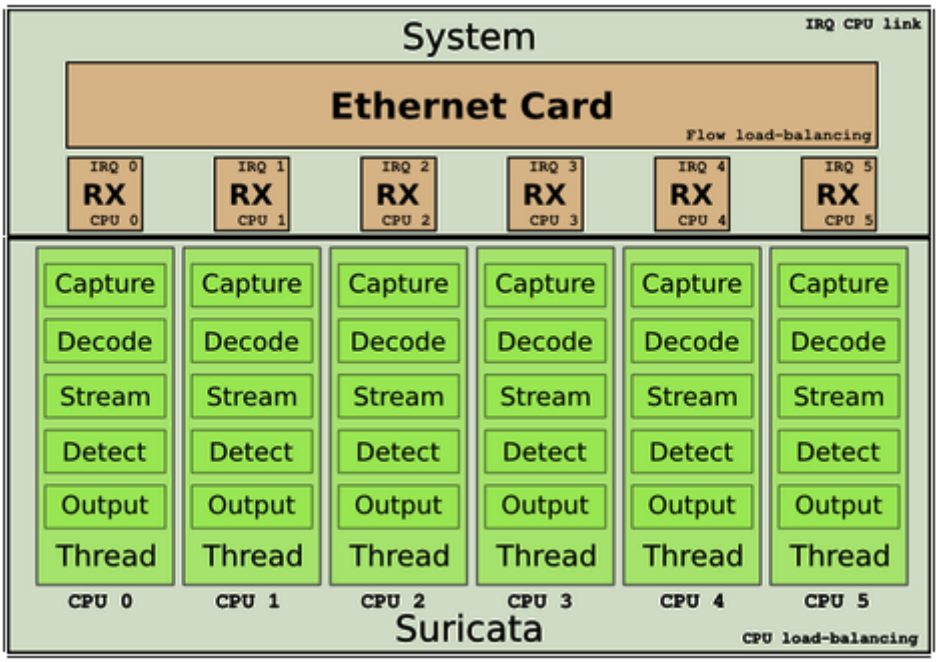
alert tcp any any -> 156.154.70.1 80 (msg:"Test Rule"; sid:5000853; content:"GET"; content:"cgi-bin/phf";)

alert -> kural eylemini belirtir; alarm ver
tcp -> hangi protokol kullanılarak gerçekleşen girişimlerde geçerli olacağı belirtilir
any -> kaynak IP adresini tanımlar, any herhangi bir IP adresi olabileceği belirtilir
any -> kaynak port, any ile herhangi bir port olabileceği belirtilmiş
156.154.70.1 -> saldırının gerçekleştiği hedef IP adresi
80 -> hedef port
msg:"Test Rule" -> Bu alarm üretildiğinde gösterilecek bilgi mesajı
sid:5000853 -> Bu kural için atanmış tanımlayıcı numara
content:"cgi-bin/phf" -> paket/mesaj içeriğinde "cgi-bin/phf" var ise bu alarm çalıştırılır.

SURICATA

Suricata açık kaynak kodlu, GPLv2 lisansı ile dağıtılan saldırı tespit ve önleme sistemidir. Kar amacı gütmeyen bir topluluk olan OISF (Open Information Security Foundation) tarafından geliştirilmekte ve desteklenmektedir. İlk olarak Aralık 2009 yılında beta sürüm, Haziran 2010'da ise ilk kararlı sürümü yayınlanmıştır.[4] Yaklaşık 10 yıl önce duyurulan ve yaygın olarak kullanılan Snort saldırı tespit sistemi gibi imza/kural tabanlı çalışmaktadır. Snort'un kullandığı kural setini desteklemesi kısa sürede kabul görmesinde etkili olmuştur.

Adını Afrika'ya özgü etobur memeli bir hayvandan (mirket) alan Suricata saldırı tespit alanında önemli yeniliklerle gelmiştir. Bunlardan ilki HTP kütüphanesi olarak adlandırılan ve Suricata proje takımından Ivan Ristic tarafından geliştirilen yeni HTTP normalizasyon aracıdır. http trafiğinin ayrıştırılmasını sağlayan bu yeni aracın en önemli özelliği "security-aware" olarak tasarlanmasıdır.[5] Yani saldırganların saldırı tespit sistemlerini atlatmak için kullanabileceği çeşitli teknikleri yakalama kapasitesine sahiptir. Bununla birlikte kütüphane http protokolüyle ilgili istek satırı, istek başlığı, URI, kullanıcı etmeni, cevap satırı, sunucu cevap satırı, çerez, "basic" ve "digest" kimlik doğrulama işlemleri için farklı ayrıştırıcılara sahiptir. Suricata'nın diğer önemli özelliği çoklu iş parçacıkları (multi-threaded) halinde çalışmayı desteklemesidir. Yani birden çok işlemci ünitesine sahip mimarilerde paket işleme işlemi farklı iş parçacıklarıyla farklı ünitelerde dağıtık olarak yapılmaktadır. Her CPU ünitesi tek iş parçacığıyla çalışan ayrı bir makine gibi davranır. Böylece yük dengesi sağlanıp, performans arttırılmış olur.[6]



Şekil 4 – Suricata'nın Multi-thread Çalışması

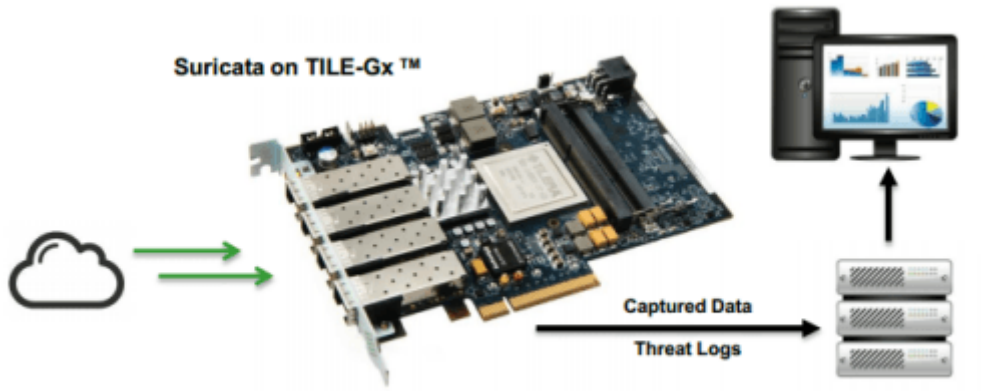
Tek iş parçacığı (single-thread) ile çalışan Snort maksimum 100-200 megabit arası trafiği işlerken, Suricata 10 gigabit gerçek trafiği işleyebilmektedir.

Özellikleri

Suricata'nın özellikleri şöyle sıralanabilir [7]:

- Saldırı tespit sistemi (IDS) , saldırı engelleme sistemi (IPS) gibi çalışma modlarında kullanılabilmektedir.
- Ağ trafiğini izleyerek trafiğin pcap formatında kaydedilmesini daha sonra kaydedilen bu dosyalarının offline olarak analiz edilmesini sağlamaktadır. Ayrıca pcap dosyalarının analizi için Unix soket modunda da çalışmaktadır.
- Linux, FreeBSD, OpenBSD, Mac OS X, Windows gibi hemen hemen tüm işletim sistemlerinde çalışabilmektedir.
- Konfigürasyon dosyası kolay bir şekilde anlaşılmayı sağlayan YAML formatındadır. Birçok programlama dili tarafından desteklenmektedir. Suricata 2.0 kararlı sürümüyle birlikte YAML dosyası istenilen parçalara ayrılarak ana dosya içerisinden çağrılması sağlanmıştır.
- IPv6 protokolü tamamen desteklenmektedir.
- Teredo, GRE, IP4-IP6 tünel protokolleri çözümlenebilmektedir.
- TCP oturumları için oturum baştan sona takip edilmesi, akışın sıraya konulması, gibi işlemleri yapar. Parçalanmaya uğrayan paketlerin yeniden bir araya getirilmesi için de ayrı bir modüle sahiptir.
- Ethernet, PPP; VLAN, QINQ vb. gibi birçok ikinci katman protokolünü desteklemektedir. Ayrıca uygulama katmanı protokollerinden HTTP, SSL, TLS, SMB, SMB2, DCERPC, SMTP, FTP, SSH, DNS çözümlenebilmektedir.
- Yazılan kurallarda PCRE (Perl Compatible Regular Expressions) kullanılabilmekte, dosya türü, boyutu, MD5 özet değeri eşleştirilmesi yapılabilmektedir.
- Çalışma sırasında yeni kural eklenmesi, silinmesi gibi kural güncelleme işlemleri yapılabilmektedir. Uygulamanın yeniden başlaması gerekmemektedir.
- NVIDIA tarafından geliştirilen ve GPU'lar tarafından kullanılan CUDA (Compute Unified Device Architecture) teknolojisini desteklemektedir. Dolayısıyla böyle bir donanım ve çoklu iş parçacıklarıyla çalışmada yüksek performans elde edilecektir.
- HTTP istekleri, TLS el sıkışmaları, SSH bağlantıları kaydedebilir. Suricata 2.0 ile birlikte DNS istek/cevapları da kaydedilmeye başlanmış ve tüm kayıtların birçok programlama dili tarafından kolayca anlaşılabilen JSON formatında kaydedilmesi sağlanmıştır.
- Kurallara göre üretilen alarmlar metin formatında kaydedilebilmekte ya da syslog'a gönderilebilmektedir. Alarmların daha hızlı kaydedilmesini sağlayan Unified2 binary formatını kullanmaktadır. Bu formattaki dosyalar Barnyard2 açık kaynak kodlu aracı kullanılarak metin haline dönüştürülebilmekte veya istenilen bir veritabanına kaydedilebilmektedir. Suricata 2.0'dan sonra HTTP isteklerinin yanı sıra Unified2 kayıtları için de XFF (X-Forwarded-For) desteği gelmiştir.
- HTTP trafiğinden geçen tüm dosyalarla ilgili bilgileri MD5 özet değerleriyle birlikte JSON formatında kaydedilebilmekte, istenildiği durumda bu dosyalar trafikten çıkarılıp belirtilen bir dizinde saklanabilmektedir.
- IPS modunda kullanılması durumunda düşürülen paketlere ilişkin bilgiler, uygulamanın çalışması ile ilgili istatistikler de kaydedilebilmektedir.

- IP itibar desteği vardır. Kural yazımında “iprep” anahtar sözcüğü kullanılarak istenilen verilerle eşleştirme yapılabilir. IP itibar desteği çalışma anında güncellenebilir, yeniden başlatma gerektirmez.
- Paket işleme performansının arttırılması için AF_PACKET, PF_RING gibi uygulamalar kullanılabilmektedir. Ayrıca Endace, Napatech, Tilera gibi özelleşmiş donanımlarda da yüksek performanslı çalışabilmektedir. 8 düğümden oluşan Tilera platformunda 80 gbps trafik Suricata ile işlenebilmektedir [8].



Şekil 5 – Suricata için Özelleştirilmiş TILE-GxDonanımı

- Suricata “Sourcefire Vulnerability Research Team™ (VRT) Rules” ve “Emerging Threats Rules” kural setleri ile uyumlu olarak çalışmaktadır. Bunun yanında Lua betik dili ile yazılacak kurallarla imzaların yetenekleri geliştirilebilir. Lua betik dili desteklenen örnek imza şöyledir [9]:

```
alert tcp any any -> any any (msg:"Lua rule"; luajit:test.lua; sid:1;)
```

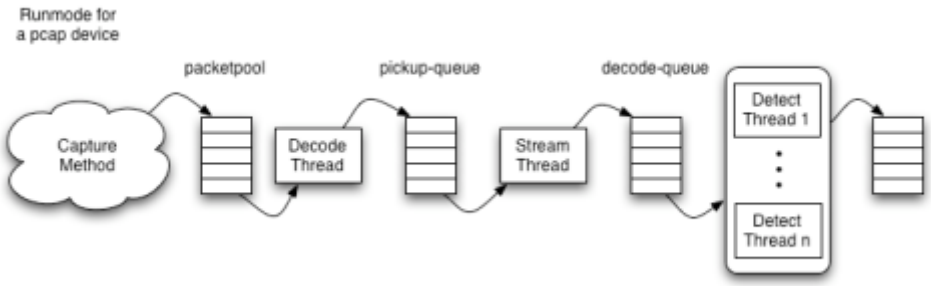
```
function init (args)
    local needs = {}
    needs["http.request_line"] = tostring(true)
    return needs
end
-- match if packet and payload both contain HTTP
function match(args)
    a = tostring(args["http.request_line"])
    if #a > 0 then
        if a:find("^POST%s+/.*%.php%s+HTTP/1.0$") then
            return 1
        end
    end
    return 0
end
```

Şekil 6 – Lua betik diliğle çalışan Suricata Kuralı

Yapısı

Birçok çalışma moduna sahip olan Suricata’nın hangi modda çalışacağı başlangıçta verilen parametrelerle belirlenmektedir. Paketlerin işlenmesi için oluşturulan kuyruk yapıları, paket işleyici iş parçacıkları çalışma modu belli olduktan sonra moda göre düzenlenerek çalışmaya uygun hale getirilir. En çok tercih edilen “pcap device” yani saldırı tespit sistemi modunda bir paket sırasıyla

paket yakalama, paket çözümleme, akış işlemi ve tespit modüllerinden geçer. Bu işlemlerin sonucuna göre paket geçirilir ya da alarm üretilir. IPS modu için paketlerin düşürülmesi ve reddedilmesi işlemleri de mevcuttur.

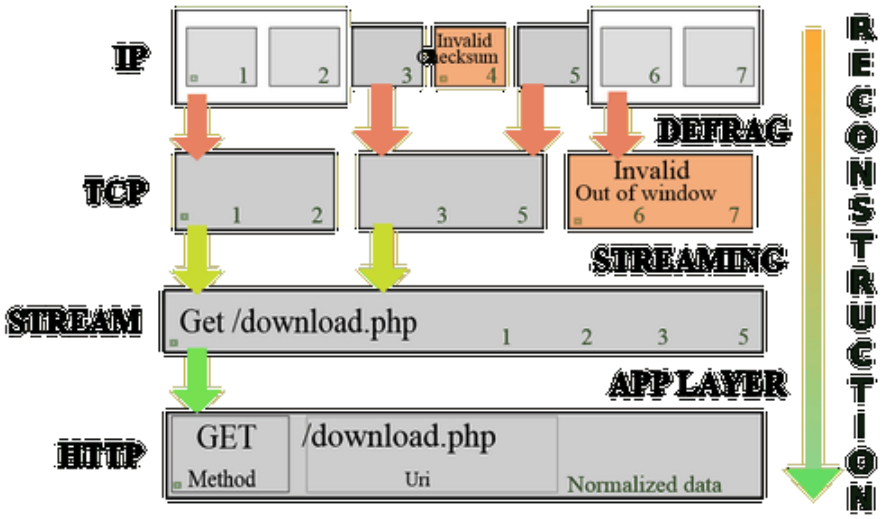


Şekil 7 – Suricata Çalışma Yapısı

Paket Çözümleme Modülü (Decoding Module): Paket çözümleme işlemi, paketlerin ara belleğe alınması ve içeriğinin Suricata'nın desteklediği veri yapısına dönüştürülmesinden sorumludur. Paketler burada veri linklerine (ethernet, ppp vb.) göre sınıflandırılıp ona uygun çözümleyicilerde işlenir [10].

Akış İşlemleri Modülü (Stream Module): Temel olarak 3 görevi vardır:

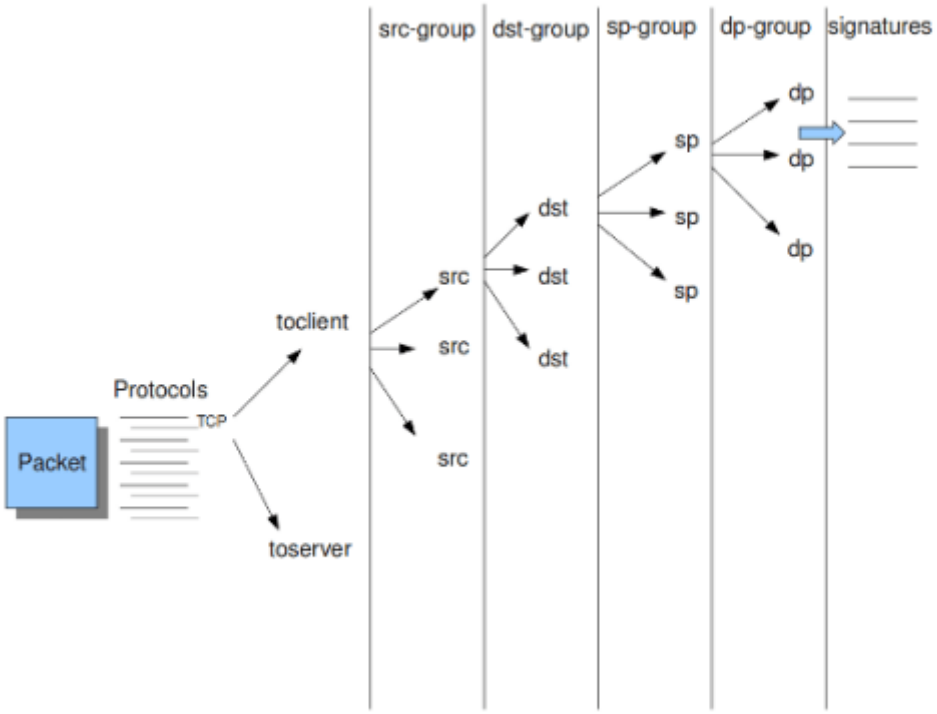
1. Doğru, anlaşılabilir bir ağ bağlantısının olması için akışları takip eder.
2. TCP bağlantıları için ana akşın tekrar oluşturulabilmesi için paketlerin sıraya konulması işlemini yapar.
3. Uygulama katmanı denetimi yapar. HTTP ve DCERPC analiz edilir.



Şekil 8 – Stream Modülü Çalışma Mekanizması

Tespit Modülü (Detect Module): Konfigürasyonda belirtilen tüm kuralların yüklenmesi, tespit eklentilerinin başlatılması ve paketlerin gruplanarak kurallarla eşleştirilmesi gibi önemli işlerden sorumludur.

Kuralları kendi içerisinde gruplandırır. Örneğin TCP paketinin UDP protokolü için yazılmış kurallarla karşılaştırılmasına gerek yoktur. BU yüzden TCP için yazılmış kurallar bir grup olarak düşünülebilir [11]



Şekil 9 – Tespit Modülü ile Paketlerin Gruplandırılması

Oluşturulacak grupların sayısı kullanıcı tarafından belirlenebilir. Grupların sayısını belirlemek bir hafıza/performans problemidir. Az sayıdaki gruplar düşük performans az bellek kullanımına neden olurken grup sayısının artması performans ve bellek kullanımının artmasına neden olur. Suricata'da tanımlı olarak "yüksek, orta ve düşük" olmak üzere 3 profil gelir, varsayılan profil bellek kullanımı ve performans arasında bir denge oluşturan "orta" dır.

Suricata İmzaları

Suricata imza tabalı bir STS olduğu için çıktı olarak üretilen uyarı, hata gibi türler sistemde tanımlı olan imzalar vasıtasıyla yapılmaktadır. İmzaların yazım kuralları Snort kurallarıyla uyumludur. Kuralların yazım kuralları ilgili bölümde detaylı bir şekilde anlatıldığından burada sadece belli suricata imzaları örnek olarak anlatılmıştır.(Ek-A)

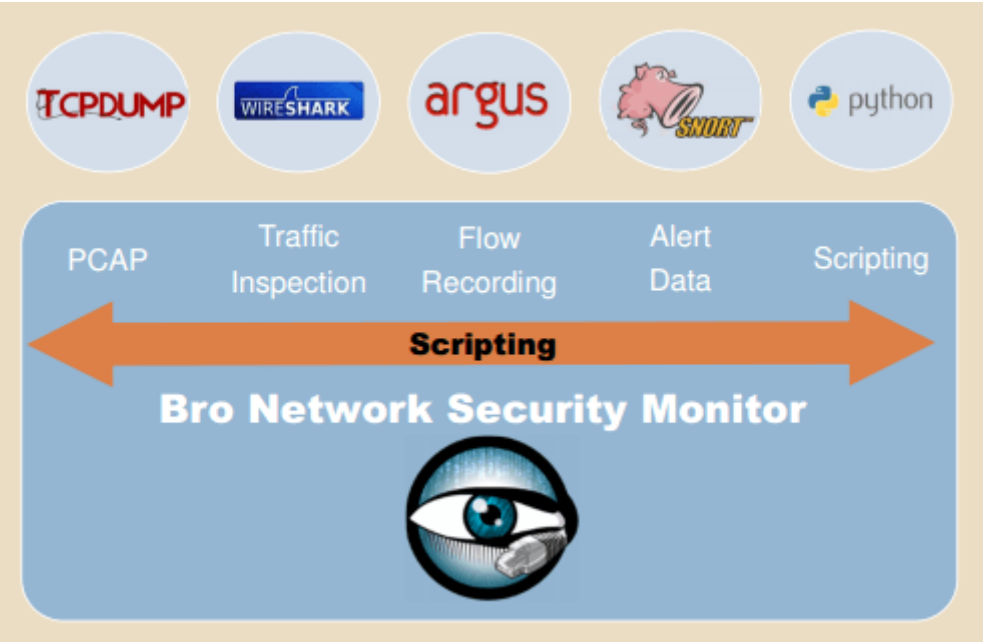
Suricata Kurulumu

Suricata kurulumu için gereken adımlar Ek-B de anlatılmıştır.

BRO

Bro açık kaynak kodlu, UNIX tabanlı, BSD lisansı ile dağıtılan saldırı tespit sistemi, ağ analiz ve izleme aracıdır. İlk olarak Lawrence Berkeley National Laboratory (LBNL)'de araştırmacı olan Vern Paxson tarafından 1995 yılında kodlanmaya başlanmıştır. 1996 yılında işlevsel olarak geliştirilmeye başlanmış ve 1998 yılında yayınlanan bir makale ile duyurulmuştur. 2003 yılına gelindiğinde National Science Foundation (NSF) tarafından proje desteklenmeye başlanmış ve günümüzde de Berkeley'deki International Computer Science Institute (ICSI)'de geliştirilmeye devam edilmektedir [12].

Bro klasik kural tabanlı IDS'lerden farklı olarak komple bir ağ trafiği analiz aracıdır. Trafik analizi sadece güvenlik alanında değil, performans analizleri ve ağ sorunlarının çözümlerini de içermektedir.



Şekil 10 – Bro'nunKapsamı

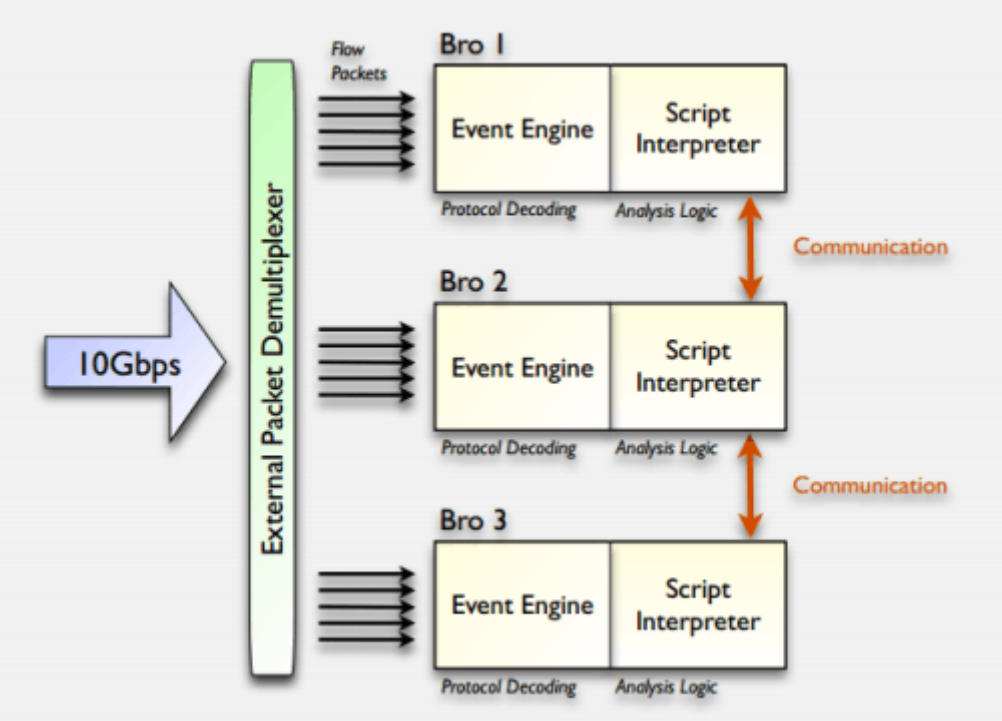
Bro çalışmasıyla birlikte ağdaki birçok aktiviteyle ilgili kayıt oluşturur. Sadece ağdaki tüm trafiğin kaydedilmesi değil özellikle uygulama katmanındaki protokollerin çözümlenmesini sağlar. Bro'yu diğer saldırı tespit sistemlerinden ayıran en önemli özellik kendine ait bir betik dilinin olmasıdır. Bu dil sayesinde çok esnek ve geliştirilebilir bir yapıdadır. Her kullanıcı yazacağı özel betiklerle sistemin fonksiyonelliğini arttırabilir ve özelleştirebilir [13]. Uygulamayla birlikte gelen birçok hazır kütüphane ve framework ile betik yazımı kolaylaştırılmıştır. Farklı yerlerde sisteme özgü Python (domain-specific Python) olarak adlandırılmaktadır. Genel olarak bu betiklerle ağdaki zararlı aktivitelerin tespiti, anomalilerin tespiti vedavranışsal analiz gibi işlemler yapılabilir. Bununla birlikte varsayılan ayarlarla da çok geniş yelpazede özellikler sunmaktadır.

Özellikleri

Bro'nun özellikleri şöyle sıralanabilir:

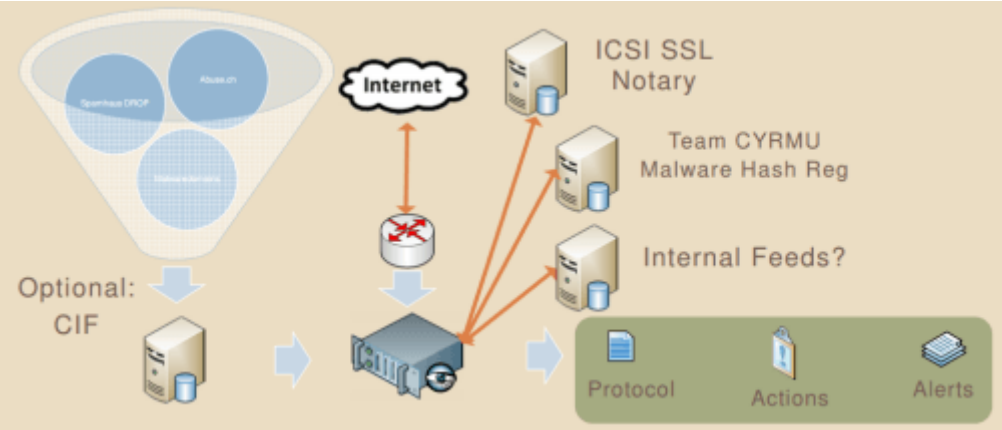
- Linux, FreeBSD, MacOS gibi UNIX tabanlı işletim sistemlerinde çalışabilmektedir.
- Gerçek zamanlı ya da offline analiz yapabilmektedir.

- Paketlerin yakalanması için “libpcap” kütüphanesini kullanmaktadır.
- Üniversiteler, araştırma laboratuvarları, büyük ölçekli işletmeler gibi trafiğin yoğun ve dağıtık olduğu yerlerde Bro kullanıcılara küme yapısını (“Bro Clusters”) sunmaktadır [14]. Farklı sunucularda Bro çalışır ve bunlar kendi arasında iletişim kurabilirler.



Şekil 11 – Bro Cluster Yapısı

- Tüm HTTP trafiğini (sunucu/istemci istek ve cevapları, mime türleri, uri vb.), DNS istek ve cevaplarını, SSL sertifikalarını, SMTP oturumlarını, FTP trafiğini çözümleyerek kaydedebilmektedir. Ayrıca ağ akışını da kayıt altına almakatadır.
- Kayıtlar rahatça okunabilir şekilde (tab karakteriyle ayrılmış), ASCII formatında metin dosyalarına kaydedilir.
- Port bağımsız olarak uygulama katmanı protokollerinden DNS, FTP, HTTP, IRC, SMTP, SSH, SSL çözümlenebilmektedir.
- HTTP, FTP, SMTP, IRC trafiğinden geçen tüm dosyalarla ilgili bilgileri MD5/SHA1 özet değerleriyle birlikte metin formatında kaydedilebilmekte, istenildiği durumda bu dosyalar trafikten çıkarılıp belirtilen bir dizinde saklanabilmektedir [15].
- Dış kaynaklar kullanarak (özet değeri eşleştirmeleri, IP itibar tabloları) çeşitli zararlı yazılımları tespit edebilmektedir [16].



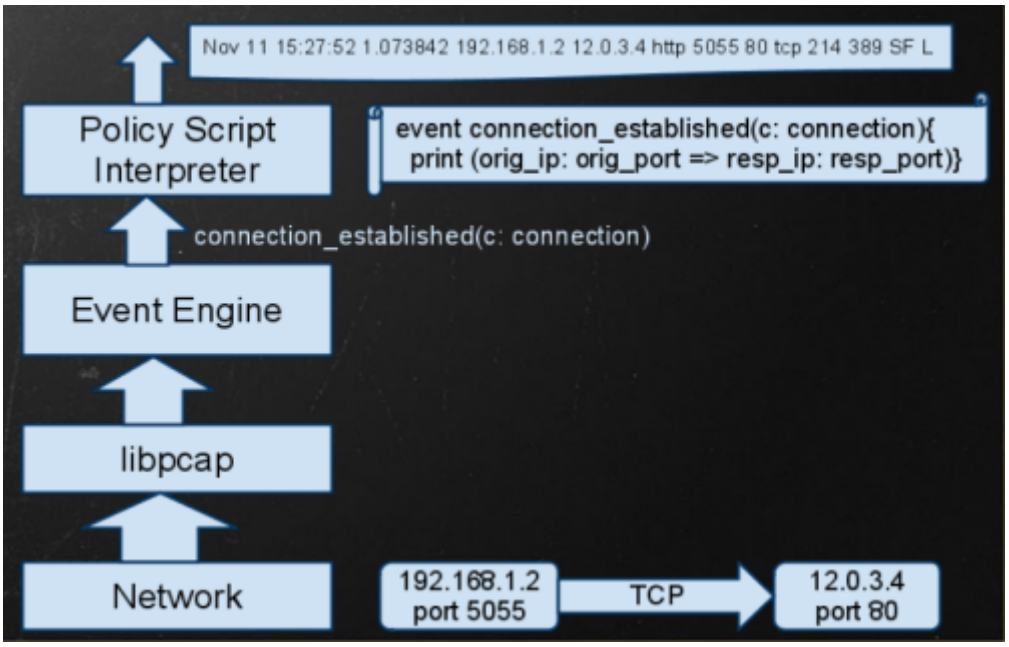
Şekil 12 – Bro'nun Dış Kaynaklarla Beslenmesi

- Ağ trafiğinde tespit edilen uygulamaların (Java, Flash vb.) açık barındıran versiyonları, popüler web uygulamaları (Skype, Facebook vb.), SSH kaba kuvvet ataklarını tespit edilebilmektedir.
- Trafikteki SSL sertifikalarına ait tüm zincirin doğrulanması sağlanmaktadır.
- IPv6 protokolü kapsamlı bir şekilde desteklenmektedir.
- Ayiya, Teredo, GTPv1 gibi tünel protokolleri tespit edilip analiz edilebilmektedir. Bro tüneli tespit ettikten sonra çözümleyerek sanki hiç tünel yokmuş gibi analiz işlemini gerçekleştirmektedir.
- Klasik IDS'lerin kullandığı desen eşleştirmesi yöntemini desteklemektedir.
- Analiz için kullanılacak dış kaynaklar gerçek zamanlı olarak sisteme entegre edilebilmektedir.
- Betik dili sayesinde tasarlanan senaryonun oluşması durumunda e-mail gönderme, anlık bağlantının sonlandırılması, yönlendirici erişim kontrol listesine blok kayıtlarının girilmesi gibi farklı bir dış işlemi tetikleyebilmektedir.
- Uygulamaların Bro ile konuşmasını sağlayan Broccoli (The Bro Client Communications Library) [17], Bro kurulumu ve kullanımı için interaktif bir kabuk sunan BroControl [18], kayıtların ayrıştırılmasını sağlayan bro-cut vb [19], snort imzalarının Bro imzalarına dönüştürülmesini sağlayan "snort2bro" betiği vb. araçlara da sahiptir.

Yapısı

Bro katmanlı bir yapıdadır ve iki temel bileşenden oluşmaktadır. Bunlar "event engine" ve "policy script interpreter"'dir.

Event Engine: C++ programlama dilinde yazılmıştır. Ağ akışındaki paket serilerini anlam ifade eden üst seviye olaylara dönüştürür. Örneğin ağdaki herhangi bir HTTP isteği IP adresleri, portları, talep edilen URI, kullanılan HTTP versiyonu ile birlikte tek bir "http_request" olayına dönüştürülür. Daha basit bir ifadeyle ağdaki herhangi bir protokole ait aktivite Bro dili tarafından anlaşılabilir formata çevrilir. Ancak buradaki örnekte HTTP isteğindeki IP ya da URI'ın zararlı olup olmadığı event engine'in görevi değildir. Bro'nun kullandığı yaklaşık 320 tane olay türü vardır [20]. Bunlardan bazıları şöyledir; new_connection, new_packet, http_header, ssl_certificate_seen, authentication_rejected, dns_PTR_reply, arp_reques.



Şekil 13 – Bro Çalışma Yapısı

Policy Script Interpreter: Bro'nun betik dilinde yazılmış olay işleyicilerinin çalıştırılmasından sorumlu yapıdır. Betikler kullanılarak ağ trafiği için oluşturulmuş olay türleri analiz edilip herhangi bir anomali olup olmadığı, olması durumunda hangi işlemlerin gerçekleştirileceği ve bunların nasıl kayıt altına alınacağı belirtilir. Daha genel bir ifadeyle trafik ile ilgili istenilen özellikler ve istatistikler elde edilebilir.

SONUÇ

Sistemleri sürekli olarak izleyebilmeyi ve saldırıları kısa süre içerisinde fark etmeyi sağlayan saldırı tespit sistemleri güvenlik sistemlerinin vazgeçilmez ürünleri arasındadır. Günümüz bilgi çağında, her kurumun internete bağlı olduğu düşünülürse herkes bir Saldırı Tespit Sistemi kurmalı ve gerekli imza güncelleştirmelerini yaparak bu sistemin çıktılarını düzenli olarak takip etmelidir diyebiliriz. Bunun firewall kadar önemli bir uygulama olduğunun, bir ihtiyaç olduğunun bilinmesi gerekmektedir. Onun için ağa gelmesi muhtemel davetsiz misafirleri fark edebilmek için maddi imkânlar ölçeğinde saldırı tespit sistemleri oluşturulmalıdır. Ticari ürünler için yeterli bütçe oluşturulamamış ise ücretsiz ürünler kullanılarak da yeterli güvenliği sağlamak çoğu zaman mümkün olabilmektedir. Sunucu tabanlı sistemlerde kurulacak port tarama saptayıcıları, dosya bütünlüğünü kontrol eden yazılımlar ya da snort, firestorm, pakemon gibi ağ tabanlı saldırı tespit sistemleri düşük maliyet ile belirli bir güvenlik seviyesi sağlamış olacaktır.

Kaynakça

[1] http://en.wikipedia.org/wiki/Morris_worm
[2] http://csrc.nist.gov/publications/history/ande80.pdf
[3] http://users.ece.cmu.edu/~adrian/731-sp04/readings/denning-ids.pdf
[4] Comparison of Open Source Intrusion Detection System
[5] Suricata: An Introduction
[6] https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Suricata_User_Guide
[7] http://suricata-ids.org/features/all-features/
[8] Tilera - Combating Cyber Attacks: Using a 288-Core Server
[9] http://workshop.netfilter.org/2013/wiki/images/1/1f/Eric_Leblond_IDS-suricata.pdf
[10] Intrusion Detection Architecture Utilizing Graphics Processors
[11] https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Suricatayaml
[12] https://www.bro.org/sphinx/intro/index.html
[13] Bro: A System for Detecting Network Intruders in Real-Time
[14] The Open Source Bro IDS Overview and Recent Developments
[15] Bro IDS File Extraction
[16] Bro IDS and the Bro Network Programming Language
[17] https://www.bro.org/download/README.broccoli.html
[18] https://www.bro.org/download/broctl.broctl.html
[19] https://www.bro.org/download/README.bro-aux.html
[20] An Overview of the Bro Intrusion Detection System

EK-A Suricata İmzaları

İmzaların çalışma davranışları aşağıdaki tabloda belirtilen parametler ile belirlenmiştir. Genel bir açıklama yapmak gerekirse: “Mesaj” sütunu imza şartları yerine geldiği durumda log dosyasında gösterilecek mesajdır. Aksiyon sütünü ise mesajın türünü içerir. Protokol, kaynak ip, kaynak port, hedef ip ve hedef port network trafiğinin iki ucunu belirtir. “Flow” sütünü network trafiğinin yönünü ve bağlantı durumunu gösterir. “İmza Detay” sütünü ise paket içeriğindeki anahtar kelimelerin varlığını kontrol eder.

İmza	**Aksiyon**	**Protokol**	**Kaynak IP**	**Kaynak Port**	**Hedef IP**	**Hedef Port**	**Mesaj**	**Flow**	**İmza Detay**
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions (Zango) Spyware TB Installer Download"; flow:to_server,established; uricontent:"/ZangoTBInstaller.exe"; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2003059; classtype:trojan-activity; sid:2003059; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions (Zango) Spyware TB Installer Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /ZangoTBInstaller.exe ifadesi varsa

alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware Actionlibs Download"; flow:to_server,established; uricontent:"/actionurls/ActionUrlb"; nocase; uricontent:"partnerid="; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2003057; classtype:trojan-activity; sid:2003057; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware Actionlibs Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /actionurls/ActionUrlb ve partnerid= ifadeleri varsa
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware config Download"; flow: to_server,established; uricontent:"/config.aspx?did="; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2002099; classtype:trojan-activity; sid:2002099; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware config Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /config.aspx?did= ifadesi varsa
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware Defs Download"; flow: to_server,established; uricontent:"/geodefs/gdf"; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2002048; classtype:trojan-activity; sid:2002048; rev:6;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware Defs Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /geodefs/gdf ifadesi varsa

alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware Install"; flow: to_server,established; uricontent:"/downloads/installers/"; nocase; content:"simpleinternet/180sainstaller.exe"; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2002003; classtype:trojan-activity; sid:2002003; rev:7;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware Install	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /downloads/installers/ ve simpleinternet/180sainstaller.exe ifadeleri varsa
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware Keywords Download"; flow: to_server,established; uricontent:"keywords/kyf"; nocase; content:"partner_id="; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2002001; classtype:trojan-activity; sid:2002001; rev:7;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware Keywords Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde keywords/kyf ve partner_id= ifadeleri varsa

alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 180solutions Spyware versionconfig POST"; flow:to_server,established; uricontent:"/versionconfig.aspx?"; uricontent:"&ver="; nocase; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2002354; classtype:trojan-activity; sid:2002354; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 180solutions Spyware versionconfig POST	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /versionconfig.aspx? Ve &ver= ifadeleri varsa
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 360safe.com related Fake Security Product Update (KillerSet)"; flow:established,to_server; content:"/?KillerSet="; fast_pattern; nocase; http_uri; content:"GET"; nocase; http_method; content:!"User-Agent|3a|"; http_header; reference:url,doc.emergingthreats.net/bin/view/Main/2008149; classtype:trojan-activity; sid:2008149; rev:8;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 360safe.com related Fake Security Product Update (KillerSet)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde GET ifadesi ve client body içinde (1st order) /?KillerSet= ifadesi bulunuyorsa ve http başlığında User-Agent: ifadesi yer almıyorsa

alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE 51yes.com Spyware Reporting User Activity"; flow:established,to_server; uricontent:"/sa.aspx?id="; nocase; uricontent:"&refe=http"; nocase; reference:url,doc.emergingthreats.net/bin/view/Main/2003620; classtype:trojan-activity; sid:2003620; rev:4;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE 51yes.com Spyware Reporting User Activity	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde &refe=http ve /sa.aspx?id= ifadeleri bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE A-d-w-a-r-e.com Activity (popup)"; flow: established,to_server; uricontent:"/cgi-bin/PopupV"; nocase; uricontent:"?ID={"; nocase; reference:url,www.a-d-w-a-r-e.com; reference:url,doc.emergingthreats.net/bin/view/Main/2001730; classtype:trojan-activity; sid:2001730; rev:9;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir.	ET MALWARE A-d-w-a-r-e.com Activity (popup)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /cgi-bin/PopupV ve ?ID={ ifadeleri bulunuyorsa

alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE Suspicious Mozilla User-Agent Likely Fake (Mozilla/5.0)"; flow:to_server,established; content:"|0d 0a|User-Agent|3a| Mozilla/5.0|0d 0a|"; nocase; content:!"|0d 0a|Host|3a| download.releasenotes.nokia.com"; content:!"Mozilla/5.0|0d 0a|Connection|3a| Close|0d 0a 0d 0a|"; reference:url,doc.emergingthreats.net/2009295; classtype:trojan-activity; sid:2009295; rev:9;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE Suspicious Mozilla User-Agent Likely Fake (Mozilla/5.0)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde "|0d 0a|User-Agent: Mozilla/5.0|0d 0a|"ifadesi olup "|0d 0a|Host: download.releasenotes.nokia.com" ve "Mozilla/5.0|0d 0a|Connection: Close|0d 0a 0d 0a|"ifadeleri yoksa (0D : carriage return, 0A: line feed)
alert http $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET MALWARE Zango Spyware (tbrequest data post)"; flow: to_server,established; uricontent:"/tbrequest"; nocase; uricontent:"&q="; nocase; pcre:"/\/tbrequest\d+\.php/Ui"; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2003610; classtype:trojan-activity; sid:2003610; rev:4;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Yaml dosyasındaki HTTP_PORTS değişkeninin değeri. Tek bir port da olabilir, liste şeklinde de olabilir. (80 ve 443 olabilir)	ET MALWARE Zango Spyware (tbrequest data post)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /tbrequest ifadesi ve &q= ifadeleri bulunuyorsa

alert http $HOME_NET any -> $EXTERNAL_NET 443 (msg:"ET MALWARE MarketScore.com Spyware SSL Access"; flow: to_server,established; content:"www.marketscore.com"; content:"InstantSSL1"; nocase; reference:url,www.marketscore.com; reference:url,www.spysweeper.com/remove-marketscore.html; reference:url,doc.emergingthreats.net/bin/view/Main/2001563; classtype:policy-violation; sid:2001563; rev:7;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	443	ET MALWARE MarketScore.com Spyware SSL Access	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde www.marketscore.com ve InstantSSL1 ifadeleri bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE /jk/exp.wmf Exploit Code Load Attempt"; flow:to_server,established; content:"/jk/exp.wmf"; nocase; http_uri; reference:url,doc.emergingthreats.net/bin/view/Main/2002999; classtype:trojan-activity; sid:2002999; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE /jk/exp.wmf Exploit Code Load Attempt	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /jk/exp.wmf ifadesi geçiyorsa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE 180solutions (Zango) Spyware Installer Download"; flow:to_server,established; content:"/downloads/valueadd/ping/ping.htm"; nocase; http_uri; content:"zango.com|0d 0a|"; http_header; reference:url,securityresponse.symantec.com/avcenter/venc/data/pf/adware.180search.html; reference:url,doc.emergingthreats.net/bin/view/Main/2003058; classtype:trojan-activity; sid:2003058; rev:6;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE 180solutions (Zango) Spyware Installer Download	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde "/downloads/valueadd/ping/ping.htm" ifadesi ve HTTP header bilgisi içinde zango.com|0d 0a|ifadesi varsa (0D : carriage return, 0A: line feed)

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE 2020search/PowerSearch Toolbar Adware/Spyware - GET"; flow:established,to_server; content:"GET"; nocase; http_method; content:"IpAddr="; nocase; http_uri; content:"&OS="; nocase; http_uri; content:"&RegistryChanged="; nocase; http_uri; content:"&RegistryUpdate="; nocase; http_uri; content:"&NewInstallation="; nocase; http_uri; content:"&utilMissing="; nocase; http_uri; content:"&Basedir="; nocase; http_uri; content:"&BundleID="; nocase; http_uri; content:"&InitInstalled="; nocase; http_uri; content:"&Interval="; nocase; http_uri; content:"&LastInitRun="; nocase; http_uri; content:"&LastInitVer="; nocase; http_uri; content:"&LastSrngRun="; nocase; http_uri; content:"&LastUtilRun="; nocase; http_uri; content:"&SrngInstalled="; nocase; http_uri; content:"&SrngVer="; nocase; http_uri; content:"&UtilInstalled="; nocase; http_uri; content:"&UtilVer="; nocase; http_uri; content:"&PCID"; nocase; http_uri; reference:url,www.sunbeltsecurity.com/ThreatDisplay.aspx?tid=13811&cs=1437A28B7A90C4C502B683CE6DE23C4E; reference:url,www.symantec.com/security_response/writeup.jsp?docid=2004-111918-0210-99; reference:url,doc.emergingthreats.net/2009807; classtype:trojan-activity; sid:2009807; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE 2020search/PowerSearch Toolbar Adware/Spyware - GET	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde GET, IpAddr=, &OS=, &RegistryChanged=, &RegistryUpdate=, &NewInstallation=, &utilMissing=, ... İfadeleri beraber bulunduğunda

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE 404Search Spyware User-Agent (404search)"; flow:established,to_server; content:"User-Agent|3a| 404search"; http_header; reference:url,doc.emergingthreats.net/2001852; classtype:trojan-activity; sid:2001852; rev:28;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE 404Search Spyware User-Agent (404search)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde "User-Agent: 404search" ifadesi bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Abcsearch.com Spyware Reporting"; flow:established,to_server; content:"/cgi-bin/search/mxml.fcgi?"; nocase; http_uri; content:"Terms="; nocase; http_uri; content:"&affiliate="; nocase; http_uri; content:"&subid="; nocase; http_uri; content:"&Hits_Per_Page="; nocase; http_uri; reference:url,doc.emergingthreats.net/bin/view/Main/2003438; classtype:trojan-activity; sid:2003438; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Abcsearch.com Spyware Reporting	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /cgi-bin/search/mxml.fcgi?, Terms=, &affiliate=, &subid=, ve &Hits_Per_Page= ifadeleri beraber bulunursa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Abox Install Report"; flow: to_server,established; content:"&time="; nocase; http_uri; content:"/new_install?id="; http_uri; reference:url,securityresponse.symantec.com/avcenter/venc/data/adware.adultbox.html; reference:url,doc.emergingthreats.net/bin/view/Main/2001441; classtype:trojan-activity; sid:2001441; rev:13;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Abox Install Report	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /new_install?id= ve &time= İfadeleri beraber bulunursa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE ABX Toolbar ActiveX Install"; flow: to_server,established; content:"/abx_search_webinstall/abx_search.cab"; http_uri; nocase; reference:url,isc.sans.org/diary.php?date=2005-03-04; reference:url,doc.emergingthreats.net/bin/view/Main/2001761; classtype:trojan-activity; sid:2001761; rev:7;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE ABX Toolbar ActiveX Install	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde /abx_search_webinstall/abx_search.cab ifadesi bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Adaware.BarACE Checkin and Update"; flow:established,to_server; content:"GET"; nocase; http_method; content:"|2E|php|3F|zone="; http_uri; nocase; content:"|26|name="; nocase; http_uri; content:"|26|bpid="; nocase; http_uri; content:"|26|bnum="; nocase; http_uri; content:"|26|pid="; nocase; http_uri; reference:url,www.symantec.com/security_response/writeup.jsp?docid=2007-021714-2431-99&tabid=2; reference:url,doc.emergingthreats.net/bin/view/Main/2008318; classtype:trojan-activity; sid:2008318; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Adaware.BarACE Checkin and Update	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde GET, .php?zone=, &name=, &bpid=, &bnum= ve &pid= ifadeleri bulunuyorsa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Adload.Generic Spyware User-Agent (91castInstallKernel)"; flow:to_server,established; content:"User-Agent|3a| 91cast"; nocase; http_header; reference:url,doc.emergingthreats.net/2003640; classtype:trojan-activity; sid:2003640; rev:11;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Adload.Generic Spyware User-Agent (91castInstallKernel)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: 91cast ifadesi bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Adload.Generic Spyware User-Agent (ProxyDown)"; flow:to_server,established; content:"User-Agent|3a| ProxyDown"; nocase; http_header; reference:url,doc.emergingthreats.net/2003639; classtype:trojan-activity; sid:2003639; rev:8;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Adload.Generic Spyware User-Agent (ProxyDown)	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde "User-Agent: ProxyDown ifadesi bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Admoke/Adload.AFB!tr.dldr Checkin"; flow: to_server,established; content:"/keyword.html"; http_uri; content:"User-Agent|3a| bdwinrun"; nocase; http_header; reference:md5,6085f2ff15282611fd82f9429d82912b; classtype:trojan-activity; sid:2008742; rev:9;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Admoke/Adload.AFB!tr.dldr Checkin	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: bdwinrun ifadesi ve request bilgisi içinde /keyword.html ifadesi varsa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Adrevmedia Related Media Manager Spyware Checkin"; flow:established,to_server; content:"User-Agent|3A| MM "; http_header; pcre:"/User-Agent\x3a MM \d\.\d+\x0d\x0a/H"; classtype:trojan-activity; sid:2013388; rev:4;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Adrevmedia Related Media Manager Spyware Checkin	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: MM ifadesi bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE adservs.com Spyware"; flow: to_server,established; content:"/binaries/relevance.dat"; http_uri; content:"adservs"; nocase; http_header; reference:url,doc.emergingthreats.net/bin/view/Main/2002740; classtype:policy-violation; sid:2002740; rev:5;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE adservs.com Spyware	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde adservs ifadesi ve request bilgisi içinde /binaries/relevance.dat ifadesi varsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Adsincontext.com Related Spyware User-Agent (Connector v1.2)"; flow: established; content:"User-Agent|3a| Connector v"; http_header; reference:url,doc.emergingthreats.net/2008372; classtype:trojan-activity; sid:2008372; rev:10;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Adsincontext.com Related Spyware User-Agent (Connector v1.2)	Üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: Connector v ifadesi varsa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE AdVantage Malware URL Infection Report"; flow:established,to_server; content:"cfg_ver="; http_uri; nocase; content:"hwd="; http_uri; nocase; content:"campaign="; http_uri; nocase; content:"ver="; http_uri; nocase; reference:url,www.siteadvisor.com/sites/config.poweredbyadvantage.com; classtype:trojan-activity; sid:2012105; rev:3;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE AdVantage Malware URL Infection Report	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP requestinin içinde cfg_ver=, hwd=, campaign= ve ver= ifadeleri bulunuyorsa
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Advertisementserver.com Spyware Checkin"; flow:to_server,established; content:"monitor.php"; nocase; http_uri; content:"?UID="; nocase; http_uri; pcre:"/UID=\d/Ui"; content:"User-Agent|3a| Microsoft URL Control"; nocase; http_header; reference:url,doc.emergingthreats.net/bin/view/Main/2007602; classtype:trojan-activity; sid:2007602; rev:8;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Advertisementserver.com Spyware Checkin	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: Microsoft URL Control ifadesi bulunuyorsa ve requestin içinde monitor.php ve ?UID= ifadeleri varsa

alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET MALWARE Advertisementserver.com Spyware Initial Checkin"; flow:to_server,established; content:"?UID="; nocase; http_uri; content:"&DIST="; nocase; http_uri; content:"&NPR="; nocase; http_uri; content:"User-Agent|3a| Microsoft URL Control"; nocase; http_header; reference:url,doc.emergingthreats.net/bin/view/Main/2007601; classtype:trojan-activity; sid:2007601; rev:6;)	Uyarı mesajı	HTTP	Yaml dosyasındaki HOME_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	Yaml dosyasındaki EXTERNAL_NET değişkeninin değeri. Tek Ip adresi de olabilir, ip bloğu da olabilir.	Herhangi bir port	ET MALWARE Advertisementserver.com Spyware Initial Checkin	Bağlantıyı başlatan taraf client ise ve üçlü el sıkışma tamamlanmışsa	HTTP header bilgisi içinde User-Agent: Microsoft URL Control ifadesi varsa ve HTTP requestinin içinde ?UID= ve &DIST= ve &NPR= ifadeleri bulunuyorsa

EK-A.2 Malware Açıklamaları

Ek-A da yer alan imza örneklerinde geçen zararlı yazılımlara ait açıklamalar aşağıdaki gibidir.

İsim	Tip	Risk Grubu	İlgili Dosyalar	Etkilediği Sistemler	Davranış
Adware.180Search	Adware	Orta	Msbb.exe Boomerang.exe 180SAInstaller.dll setup4156.exe sac.exe sau.exe 1802.dll salmbundle	Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows Server 2003, Windows XP	Web browser uygulamalarının içeriklerini takip eden bir programdır. Uygulama ekranında belirli anahtar kelimeler gördüğünde partner sitelere yönlendirme yapar. Ayrıca Adware.180Solutions ile ilgili dosyaları sisteme yükler.
Adware.ABXToolbar	Adware	Yüksek	ABX_Search.dll	Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows Server 2003, Windows XP	Bu uygulama yüksek sayıda pop-up ekranlar oluşturan web browser yardımcı objesidir.
Adware.Adultbox	Adware	Düşük	Abox.exe	Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows Server 2003, Windows XP	Yetişkin içerikli sitelere ve reklamlara ulaşım öneren bir uygulamadır.
Adware.BarACE	Adware	Düşük	-	Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows XP	Adware.BarACE adware türünde bir uygulamadır. Kullanıcı PCTurbo yu yüklerken ona danışmadan kendini toolbar olarak yükler ve bazı anahtar kelimeleri dışarıda bulunan bir url adresine gönderir.
Spyware.PowerSearch	Spyware	Yüksek	pwrsdemo.dll,gamebar.dll	Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows Server 2003, Windows XP	PowerSearch, internet explorer toolbar olarak kullanıcının sirsearch.com aracılığıyla arama yapmasına izin veren bir uygulamadır. Kullanıcının bilgilerini izinsiz olarak toplayarak onların alışkanlıklarına göre kendi reklamlarına yönlendirir.

EK-B Suricata Kurulumu

Suricata uygulamasının Ubuntu 14 Server versiyonuna kurulması için gereken adımlar aşağıda listelenmiştir.

1. Suricata uygulamasınının çalıştırılabilir şekilde üretilmesi ve çalışması için gerekli olan yardımcı programlar indirilir.
   - sudo apt-get install build-essential automake libtool bison subversion pkg-config
   - sudo apt-get install libxml2-dev libxslt-dev autoconf libc6-dev ncurses-dev libpcre3 libpcre3-dev
   - sudo apt-get install openssl libreadline6 libreadline6-dev curl git-core zlib1g zlib1g-dev libssl-dev libyaml-dev libsqlite3-dev sqlite3
   - sudo apt-get install libnet1 libnet1-dev
   - sudo apt-get install libpcap-dev libpcap0.8 libpcap0.8-dev
   - sudo apt-get install libcap-ng-dev
   - sudo apt-get install coccinelle
   - sudo apt-get install libcap-ng-dev
   - sudo apt-get install magic libmagic-dev
   - sudo apt-get install file
   - sudo apt-get install libjansson4 libjansson-dev python-simplejson
2. Suricata uygulaması http://suricata-ids.org/download/ adresinden indirilir. Örnek olarak suricata-2.0.4.tar.gz versiyonunu indirilebilir.
   - wget http://www.openinfosecfoundation.org/download/suricata-2.0.4.tar.gz
   - tar zxvf suricata-2.0.4.tar.gz
   - cd suricata-2.0.4
3. Ardından indirilen kodun çalıştırılabilir obje kodlarına çevrilmesi ve yüklenmesi gerekir.
   - ./configure --prefix=/opt/suricata --sysconfdir=/opt/suricata/etc --localstatedir=/var
   - make -j4 <<-- 4 çekirdek, istenilen çekirdek sayısı girilebilir varsayılan için boş bırakılabilir.
   - sudo make install-full
4. Ardından arayüz ayarları eklenir.
   - sudo ethtool -k eth0
   - sudo ethtool -K eth0 tx off rx off sg off gso off gro off
5. Suricata programı çalışmaya hazırdır. Servis haline döndürmek için /etc/init/suricata.conf dosyası oluşturulup aşağıdaki satırlar oluşturulan dosyaya eklenir:

   # suricata
   description "Intruder Detection System Daemon"
   start on runlevel [2345]
   stop on runlevel [!2345]
   expect fork
   exec /opt/suricata/bin/suricata -D --pidfile /var/run/suricata.pid -c
   /opt/suricata/etc/suricata/suricata.yaml --af-packet=eth0

6. Suricata uygulaması aşağıdaki komut ile başlatıp durdurulabilir.
    - sudo service suricata start/stop

Suricata'ya yeni bir imza eklemek için aşağıdaki işlemler yapılmalıdır.

1. /opt/suricata/etc/suricata/rules klasörü altında içine istenilen kuralların yazılabileceği local.rules isimli bir dosya oluşturulur.
2. /opt/suricata/etc/suricata/suricata.yaml dosyasının içinde rule-files girdisinin altındaki kural dosyaları listesine oluşturulan dosya eklenir.
3. Local.rules dosyasının içeriği değiştirildiğinde suricata uygulaması yeniden başlatılmalıdır. Bu durumda eklenilen yeni kurallar sisteme eklenmiş olacak ve alarm üretmeye başlayacaktır. Örnek olarak aşağıdaki imza eklenebilir. İmza herhangi bir ip ve port kısıtlaması olmaksızın ICMP paketlerine karşı "PING detected" mesajı üretmektedir.
    - alert icmp any any -> any any (msg:"PING detected"; sid:2; rev:1;)

Uygulama imza tabanlı olduğu için imzaların sürekli güncellenmesi gerekmektedir. Bunu otomatik sağlayacak olan OinkMaster yada PulledPork uygulamalarının konfigürasyonu incelenebilir. Ayrıca imza sayısı ve network trafiği ile doğru orantılı olarak log kayıtlarının da HDD de fazla yer tutmaması için logrotate yazılımı konfigüre edilebilir.