Tehdit Risklerini Modelleme

Uygulamanızı tasarlarken, en çok önem vermeniz gereken tehdit riski tayin edilmiş kontrolleri kullanmaktır. Aksi halde kaynaklarınızı, zamanınızı ve paranızı gereksiz kontroller için boş yere harcarsınız ve bu gerçek riskler için yeterli değildir.

Riskleri belirlemek için kullandığınız metot, yapılandırılmış tehdit riskleri modellemesi yapmanız kadar önemli değildir. Microsoft, güvenlik geliştirme programının en önemli ilerlemesinin, tehdit modellemesine evrensel bir uyum sağlaması olduğunu belirtmiştir.

OWASP, Microsoft'un tehdit modelleme yöntemini, uygulama güvenliğinde nadir olarak karşılaşılan sorunları iyi çözebilmesi ve tasarımcılar, geliştiriciler ve kodu kontrol eden kişiler tarafından kolay uyum sağlanıp öğrenilmesinden dolayı seçmiştir.

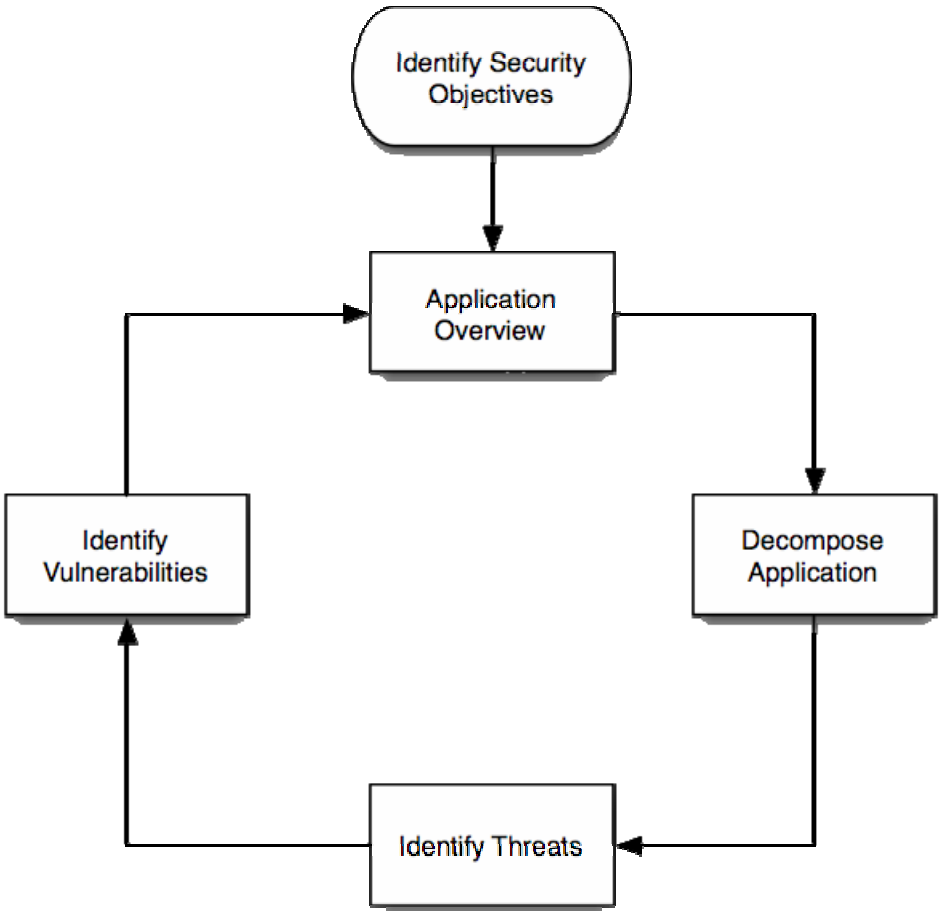
Microsoft Tehdit Modelleme Yöntemi Kullanılarak Tehdit Risklerini Modelleme

Tehdit modelleme, güvenli web uygulamalarının gelişimi için temeldir ve organizasyonların doğru kontrolleri belirlemesini ve bütçe içinde etkili karşı tedbir oluşturmasını sağlar. Örneğin; ihmal edilebilecek kadar hilesi olan bir sisteme 100,000 dolarlık bir kontrol eklemek.

Tehdit Riskleri Modellemesinin Yapılması

Tehdit modelleme yönteminde beş aşama bulunmaktadır. Microsoft tehdit ağaçlarını takip etmeyi ve göstermeyi destekleyen ,.NET'te yazılmış bir tehdit modelleme aracı sağlar. Bu aracın kullanımı büyük ve uzun ömürlü projeler için yardımcı olabilir.

Tehdit Model Akışı



Güvenlik Hedeflerinin Belirlenmesi

Şirketler (veya topluluk başkanlıkları) güvenlik hedeflerinin belirlenmesinde geliştirici takım ile hem fikir olmalıdır. Uygulamaların güvenlik hedefleri aşağıdakilere ihtiyaç duyar:

- Kimlik: Bu uygulama kullanıcının kimliğini suiistimallere karşı koruyor mu? Kimliğin doğruluğuna emin olmak için yeterli kontroller var mı? (Birçok banka uygulaması için gereklidir.)
- İtibar: Suiistimal edilmiş veya saldırıya uğramış uygulamaların neden olduğu itibar kaybı
- Mali: Risk seviyesindeki şirketler potansiyel mali kayıpları iyileştirmek için

yatırım yapmalıdırlar. Forum yazılımları ortak internet bankacılığından daha az mali risk taşır.

- Gizlilik ve düzenleme: Kullanıcının bilgilerinin ne kadar kapsamlı korunduğudur. Forum yazılımı yapısı gereği herkes tarafından görülebilir, fakat bir vergi programı birçok ülkede vergi düzenlemeleri ve gizlilik kanunlarıyla sınırlandırılmıştır.
- Kullanılırlık garantileri: Bu yazılım SLA veya benzer bir anlaşma tarafından tanınması gerekli midir? Ulusal olarak korunan bir altyapısı var mı? Hangi aşamada uygulamayı kullanmak gereklidir? Çok kullanılan uygulamalar ve teknikler aşırı derecede pahalıdır. Bu yüzden, doğru kontrollerin kurulması büyük ölçüde zaman, kaynak ve para kaybını önler.

Bu ayrıntılı bir liste sayılamaz fakat kurulmuş teknik kontrolleri yürüten şirketin risk kararlarına dair bir fikir verir. Risk kılavuzunun diğer kaynakları bunlardır:

- Yasalar(gizlilik yasası veya mali yasalar gibi)
- Düzenlemeler(bankacılık veya e-ticaret düzenlemeleri gibi)
- Standartlar (ISO 17799 gibi)
- Yasal Anlaşmalar (ticari anlaşmalar gibi)
- Bilgi Güvenliği Poliçesi

Uygulamanın Gözden Geçirilmesi

Güvenlik hedefleri tanımlandığında, aşağıdakileri belirlemek için uygulama analiz edilmeli:

- Uygulamanın Kısımları
- Veri Akışı
- Güvenlik Sınırları

Bunu yapmanın en iyi yolu uygulamanın yapısı ve tasarım dokümanlarını incelemektir. Uygulamanın kısımlarının UML şemalarına bakılmalıdır. Uygulamanın en üst kısmının şemaları genellikle veri akışının çeşitli yerlere neden ve nasıl olduğunun anlaşılması için gereklidir. Güvenlik sınırı ile çakışan veri(İnternette kaynak kod veya şirket mantığında

veritabanı sunucusu gibi) dikkatlice analiz edilmeliyken aynı güvenlik seviyesinde akan verinin tekrar kontrol edilmesine gerek yoktur.

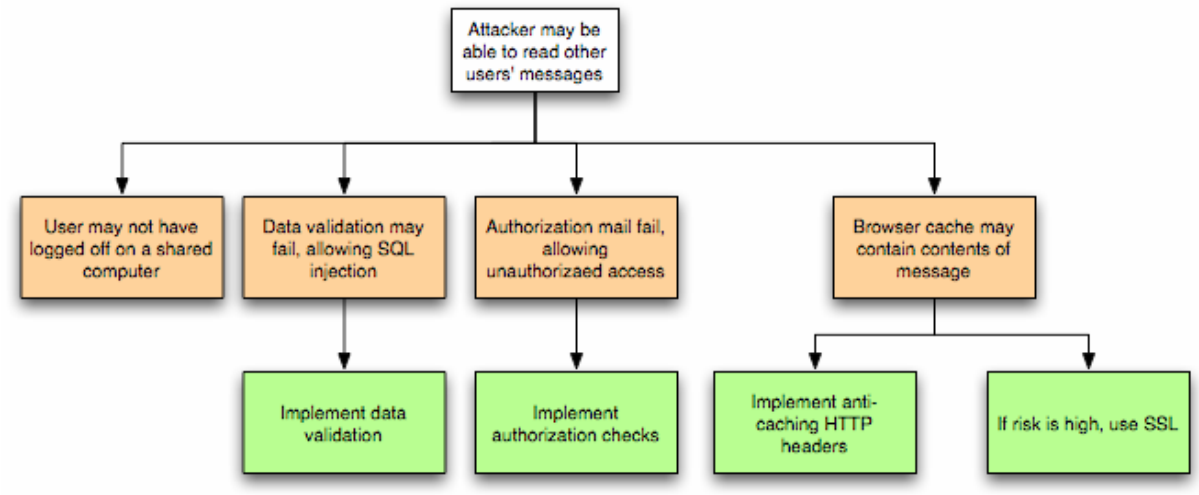
Uygulamayı Ayrıştırma

Uygulamanın yapısı anlaşıldığında, uygulama ayrıştırılmalıdır. Bunun anlamı güvenlik etkisine sahip bütün özelliklerin ve kısımların ayrıştırılmasıdır. Örneğin; doğruluğu kanıtlanmış bir kısım araştırılırken, verinin bu kısma nasıl girdiği, bu kısmın veriyi nasıl onayladığını, verinin akışı ve eğer veri kaydediliyorsa bu kısmın aldığı kararların neler olduğu anlaşılmalıdır.

Bilinen Tehditlerin Dokümanı

Bilinmeyen tehditleri listelemek imkânsızdır ve talihsizce yeni savunmasızlıklara karşı koymak için birçok müşteri sistemi kötü yazılımlar (malware) üretmiştir. Bunu yerine, bilinen ve araçlarla ya da Bugtraq'tan kolayca belirlenebilen riskler üzerinde yoğunlaşılmalıdır.

Bir tehdit yazılırken Microsoft iki farklı yaklaşım önerir. Bunlardan biri tehdit grafiği, diğeri ise yapı listesidir. Tipik olarak, tehdit grafiği okuyuculara kısa sürede birçok bilgi verir, fakat yapılandırılması uzun sürer. Yapı listesi yazmak için daha kolaydır fakat tehdidin etkisinin açığa kavuşmasında daha uzun süre alır.



Şekil 1: Grafiksel tehdit ağacı

1. Saldırgan diğer kullanıcıların mesajlarını okuyabilir
   * Paylaşılan bir bilgisayarda kullanıcı sistemden çıkmamış olabilir.
2. Veri doğrulanması SQL girişine izin verebilir.

3. Veri doğrulanmasını gerçekleştirme
4. Yekti sistemi başarısız olabilir, yetkisiz ulaşıma izin verme
5. Yetki kontrolünü gerçekleştirme
6. Tarayıcı önbellek mesajın içeriğine sahip olabilir.
7. Önbelleğe alınmayan http başlıklarını gerçekleştirme
8. Eğer risk yüksekse SSL kullanımı

Tehditler saldırganlar tarafından harekete geçirilir; onlar genellikle sizin uygulamalarınızdan veya önlenmiş kontrollerinizden bir şeyler isterler. Hangi tehditlerin uygulanabilir olduğunu anlamak, uygulamaya kimin saldırdığını anlamak için güvenlik hedeflerini kullanmak:

- Rastlantısal buluş: yetkili kullanıcılar uygulama mantığında tarayıcı kullanarak hataya düşerler.
- Otomatikleştirilmiş kötü yazılım (malware) (bilinen savunmasızlıkları fakat bu kez biraz daha dikkatli ve akıllıca araştırmak)
- Meraklı saldırgan (sizin uygulamanızda bir şeylerin yanlış gittiğini fark eden güvenlik araştırmacıları veya kullanıcıları gibi)
- Betik Hırsızlığı: bilgisayar suçluları “saygı” için veya politik sebepler yüzünden uygulamalara saldırır ve ya uygulamaları bozar- kullanılan teknikler burada ve OWASP test kılavuzunda uygulamayı uzlaştırmak için açıklandı
- Harekete geçmiş saldırganlar(huzursuz personel ve ücretli saldırganlar gibi)
- Organize suç (genellikle e- ticaret veya bankacılık gibi yüksek riskli uygulamalar)

  Karşı koyduğun saldırganların seviyesini bilmek büyük önem taşır. Örneğin, işlemlerinizi anlayan bilgili bir saldırgan bir betik hırsızından çok daha fazla tehlikelidir.

STRIDE

Yanıltıcı kimlik

Yanıltıcı kimlik birçok kullanıcının sahip olduğu, uygulamanın anahtar risklerinden biridir. Fakat yanıltıcı kimlik uygulamada bir tane yürütme içeriği ve veritabanı seviyesi kullanır. Kullanıcılar kesinlikler başka bir kullanıcı gibi davranmamalı veya o kullanıcının yerine geçmemelidir.

Veriyle oynamak

Kullanıcılar onlara ulaşan herhangi bir veriyi değiştirebilirler ve böylece alıcı doğrulamasını, GET ve POST verisini, HTTP başlıklarını ve bu gibi şeyleri değiştirebilirler. Uygulama, kullanıcıya uygulamanın kendisi tarafından bulunabilen faiz oranları veya dönemleri gibi verileri göndermemelidir. Uygulama, kullanıcıdan gelen herhangi bir verinin uygulanabilirliğini ve mantıklı olup olmadığını dikkatlice kontrol etmelidir.

Reddetme

Yetersiz izlenilebilirlik ve kullanıcı aktivitelerinin yetersiz denetlenmesi söz konusu ise kullanıcı işleme itiraz edebilir. Örneğin, eğer kullanıcı " Parayı bu dış hesaba aktaramadım" derse ve başından sonuna kadar onların aktivitelerini izleyemediyseniz, bu kesinlikle işlemin başarısız olduğu anlamına gelir.

Uygulamalar yeterli reddetme kontrolüne sahip olmalıdır, web erişim günlükleri, her kattaki izlerin kontrolü ve başından sonuna kadar kullanıcı içeriği gibi. Tercihen uygulama kullanıcı tarafından çalıştırılmalıdır fakat bu birçok çatıda (framework) mümkün olmamaktadır.

Bilgi Açıklama

Kullanıcılar özel detayları siteme girmekten sakınırlar. Eğer bir saldırganın, kullanıcının detaylarını açıklaması mümkün olursa, isimsiz veya yetkili bir kullanıcı olsa da, itibar kaybının olacağı bir süreç başlayacaktır. Uygulamalar, kullanıcılar tek bir üyelikle tüm bir programı çalıştırırken, kullanıcı ID' siyle oynanmasını engellemek için güçlü kontroller içermelidir.

Kullanıcının tarayıcısı bilgi sızdırabilir. Her tarayıcı, HTTP başlıkları tarafından istenen ön belleğe alınmama yönergesini uygulamaz. Her uygulama, bilginin sızdırılmaması ve bir saldırgan tarafından daha çok bilgi öğrenebilmek için kullanılamaması için tarayıcı tarafından kaydedilen bilginin miktarını minimuma indirmelidir.

Servis reddi

Uygulamalar servisin reddettiği saldırılar tarafından kötü kullanılabileceklerinden haberdar olmalıdırlar. Onaylanmış uygulamalar için pahalı kaynaklar geniş dosyalar, karışık hesaplamalar, ağır görev araştırmaları veya isimsiz kullanıcılar için değil yetkili kullanıcılar için muhafaza edilmesi gereken sorgular gibi. BU lükse sahip olmayan uygulamalarda, en az işi yapmak için uygulamanın bütün kesimleri yürütülmeli, hızlı veritabanı sorguları kullanılmalı ya da hiç kullanılmamalı, geniş

dosyalar açığa çıkarılmamalı veya servisin reddettiği basit bir saldırıyı engellemek için her kullanıcıya özel bağlantılar (links) sağlanmalı.

Ayrıcalık Seviyesi

Eğer bir uygulama kullanıcı ve yönetici işlevleri sağlıyorsa, kullanıcının herhangi yüksek bir seviyenin işlevlerine erişemeyeceğinden emin olmak önemlidir.

Özellikle, sadece kullanıcı bağlantıları(link) sağlamamak yeterli değildir – ayrıcalıklı işlevselliklere sadece doğru kişilerin ulaşabildiğinden emin olmak için bütün eylemler bir otorite matrisinden geçmelidir.

DREAD

DREAD risk oranının altındaki düşüncenin bir parçasını oluşturmak ve riskleri sınırlandırmak için kullanılır.

DREAD beş elementin ortalaması olan risk değerini hesaplamak için kullanılır.

$Risk_{DREAD}$ = (DAMAGE + REPRODUCABILITY + EXPLOITABILITY + AFFECTED USERS + DISCOVERABILITY) / 5

Bu 0 ile 10 arasında bir rakam ortaya çıkarır. En yüksek rakam en ciddi risktir.

Zarar Potansiyeli (DAMAGE)

Eğer tehdit gerçekleşirse ne kadar zarara neden olur?

0 = Hiç	5 = bireysel kullanıcı bilgisi uzlaşmış veya etkilenmiş	10 = bütün sistemin yıkımı

Üretilebilirlik (REPRODUCIBILITY)

Bu tehdidi üretmek ne kadar kolay?

0 = uygulamanın idarecileri için bile çok zor veya imkânsız	5 = bir veya iki aşama gerekir belki yetkili bir kullanıcı da gerekebilir	10 = sadece bir tarayıcı gerektirir ve başlamamış adres çubuğu

Yararlanılabilirlik (EXPLOITABILITY)

Bu tehditten yararlanmak için neye ihtiyaç vardır?

0 = gelişmiş programlama ve şebeke yeteneği, gelişmiş veya	5 = kötü yazılımın varlığı (malware) veya kolayca	10 = sadece bir tarayıcı

müşteri saldırı araçları	kullanılabilecek normal saldırı araçları	

Etkilenmiş kullanıcılar (AFFECTED USERS)

Bu tehdit kaç kullanıcıyı etkilemiştir?

0 = Hiç	5 = bazı kullanıcıları fakat hepsini değil	10 = Bütün kullanıcıları

Keşfedilebilirlik (DISCOVERABILITY)

Bu tehdidi keşfetmek ne kadar kolay? Var olan bir uygulamada kodlar gözden geçirilirken, bu konuların keşfedildiği farz etmek için "Keşfedilebilirlik" 10'a eşitlenir.

0 = olmama ihtimali çok zor. Kaynak ve sistem ulaşımı gerektirir.	5 = Tahmin edilerek veya şebeke ağları takip edilerek kavranabilir	9 = Hataların detayları genel alanda (public domain) ve Google kullanılarak keşfedilebilir. 10 = Bir adres çubuğunda veya bir şekle sahip

Alternatif tehdit modelleme sistemleri

OWASP Microsoft modelleme işlemine uyum sağlamanın bazı topluluklarda tartışmalı bir seçim olduğunu onayladı. Eğer asılsız önyargılarla STRIDE / DREAD kabul edilemez oluyorsa, biz her topluluğun var olan uygulama ve tasarımlara karşı değişik tehdit modellerini denemesini öneriyoruz. Bu topluluğa hangi yaklaşımın onlar için en iyisi olduğuna karar vermesine ve onların topluluğu için en uygun tehdit modelleme araçlarına uyum sağlamasına izin verir.

Tehdit modellemesi yapmak bu kılavuzdaki diğer kontrollerden çok daha büyük bir dönüş sağlar. Bu tehdit modellemesinin meydana geldiği yerler için çok önemlidir.

Trike

Trike, Microsoft tehdit modelleme yöntemine benzerlikleri olan bir tehdit

modelleme çatısıdır(framework) fakat Trike, STRIDE/DREAD ile gösterilen karışık tehdit modelleri kullanmak yerine(saldırılar, tehditler, zayıflıklar) ayrı uygulama, tehdit ve risk modelleri kullanır.

Trike kağıdında Trike'ın amaçları şunlardır:

> Tarafların yardımıyla,
>
> Sistem üyelerinin desteğiyle, her bir envanter için riskin tüm üyeler tarafından kabul edilebilir olduğundan emin olunması
>
> Eğer bunu yapmadıysak, söyleyebilmek.
>
> Neler yaptığımızı ve etkilerini taraflara iletmek.
>
> Taraflara, kendilerinin ve diğer tarafların eylemleriyle kendi alanlarda içerdikleri riskleri anlamak ve azaltmak için yetki vermek.

Daha fazla bilgi için lütfen aşağıdaki "Ek Okumalar" bölümünü gözden geçiriniz. OWASP Guide 2.1 (due November 2005) Trike hakkında daha fazla bilgi verecektir.

AS/NZS 4360:2004 Risk Yönetimi

1999'da ortaya çıkan Avustralya Standardı / Yeni Zelanda Standardı AS/NZS 4360 riskleri yönetmede ve belgelemede Dünyanın ilk resmi standardıdır ve hala riskleri yönetmedeki nadir standartlardan biridir. 2004 yılında yeniden yapılandırılmıştır.

AS/NZS 4360 kolay (sadece 28 sayfa uzunluğunda) ve esnektir ve risk yönetimi AS/NZS 4360'nın beş basamağını gerçekleştiği sürece toplulukları belli bir risk yönetim metodu için zorlamaz.

AS/NZS 4360'nın tekrarlanan beş önemli kısmı:

İçeriği saptamak – ele alınan risklerin saptanması diğer bir deyişle hangi nitelikler/sistemler önemli

Risklerin belirlenmesi – Ele alınan sistemin içinde hangi riskler görülmüştür?

Risklerin analizi – risklere bakmak ve eğer destekleyici kontroller varsa bunları belirlemek

Riskleri hesaplama – artan riskleri belirleme

Riskleri ele alma – şirketler tarafından seçilen risklerin azaltılması için riskler ele alınırken izlenen metodun tarifi

AS/NZS 4360 risklerin işlevsel risk stili grubu tarafından yöneltileceğini varsayar, bu topluluk yeterli yeteneklere sahiptir ve risk yönetimi grupları riskleri belirler, analiz eder ve ele alır.

Neden AS/NZS 4360'ı kullanmalısınız:

AS/NZS 4360, Sarbanes-Oxley uyumu gerektiren organizasyonlar için, risk yönetim metodolojisi olarak iyi çalışır.

AS/NZS 4360, riskleri yönetmek için ihtimal ve sonuç gibi geleneksel bir yöntem kullanmayı tercih eden organizasyonlar için iyi çalışır.

AS/NZS 4360 dünya çapında birçok risk yöneticisine tanıdık ve sizin organizasyonunuz AS 4360'e uyumlu bir yaklaşımla yürütülüyor olabilir.

Eğer bir Avustralya organizasyonu iseniz ve düzenli esaslarla denetleniyorsanız bunu kullanmanız gerekli olabilir veya neden kullanmadığınızı açıklayın. Neyse ki, yukarda anlatılan STRIDE / DREAD modeli AS/NZS 4360 ile uyumlu.

Neden AS/NZS 4360'ı kullanmamalısınız:

AS/NZS 4360 yaklaşımı şirket veya sistemik riskler için teknik risklerden daha iyi çalışır.

AS/NZS 4360 tehdit risk modeli yapılı uygulamaları(exercise) sağlayan metotları ele almaz.

AS/NZS 4360 risklerin yönetimi için genel bir çatı(framework) olarak, web uygulamalarının güvenlik risklerini belirten hiçbir yapısal metot sağlamaz.

AS 4360 güvenlik incelemeleri için riskleri sınıflandırabilse de web uygulamalarındaki tehditleri belirten yapısal metottaki eksikliği onu diğer metotlardan daha az çekici kılar.

Neden CVSS kullanılmalı:

Eğer güvenlik araştırmacılarından veya diğer kaynaklardan sizin ürününüzün savunmasızlığına dair bir bildiri aldıysanız, siz yamaları serbest bıraktığınızda kullanıcılarınızın uygun seviyede aktivite yapmaları gerekliliği hakkında uyarmak için emniyetli bir güvenlik oranına sahip olduğundan emin olmak istersiniz.

Eğer bir güvenlik araştırmacısı iseniz ve programınız içinde birçok sömürü bulduysanız, gerçekçi risk oranları üretebilmek ve ISV'nin bu sömürüleri kendi oranlarıyla karşılaştıracağından emin olmak için CVSS sıralama sistemini kullanmak istersiniz.

CVSS ABD'nin çalışma grubu tarafından tavsiye edilir. -Bu dokümanın yazıldığı sırada hala politikanın bu olup olmadığı belirsizdir.

Neden CVSS kullanılmamalı:

CVSS saldırı bölgesini bulamaz ve ya azaltamaz (tasarım hatası) ya da kodun rastgele seçilmiş bir parçasından olası risklerin sıralanmasına bu amaçla

tasarlanmadığı için yardım edemez.

CVSS, STRIDE / DREAD'dan çok daha karmaşıktır. Çünkü CVSS'nin amacı serbest bırakılan yazılıma uygulanan ilan edilen savunmasızlığın risklerini modellemektir.

CVSS risk sıralaması karmaşıktır – bir hesap çizelgesinin riskleri hesaplaması gerekir çünkü CVSS'nin arkasındaki varsayım küçük bir riskin ilan edilebileceği veya küçük saldırı vektörlerini amaçlayan bir kurt (worm) ya da Trojan'ın serbest bırakıldığıdır.

Eğer 250 veya daha fazla tehdidin sıralanabileceği bir kod gözden geçirilirse, hesaplamanın üstünde CVSS risk sıralaması biraz daha yüksektir.

Octave

OCTAVE, CMU Yazılım Mühendisliği Enstitüsünde CERT ile birlikle işbirliğiyle oluşturulmuş ağırsıklet risk yöntemi yaklaşımıdır. OCTAVE teknik riskleri değil organizasyonla ilgili riskleri hedeflemiştir.

OCTAVE 'in iki sürümü vardır: OCTAVE – büyük organizasyonlar için ve OCTAVE-S – küçük organizasyonlar için, ikisi de pratik, görüntü ve çalışma tablosu kataloguna sahiptir. OCTAVE, birçok siteyle popüler olmuştur.

OCTAVE ne zaman kullanışlı:

Bir organizasyonun içinde risk yönetim ve kontrol kültürü oluşturulurken

İş riskini ölçerken ve belgelerken

Tüm IT güvenlik riskini ölçerken ve belgelerken, özellikle IT risk yönetim şirketinin tamamıyla ilgiliyse

Bütün sistemleri kapsayan riskleri belgelerken

Bir organizasyon tam olduğunda, alanda çalışma riski yöntemi yoktur ve alana koymak için güçlü bir risk yönetim çatısı(framework) gerektirir

OCTAVE'in başarısızlıkları:

OCTAVE, AS 4360 ile uyumsudur, zorlamayla ihtimal=1 (bir tehdit her zaman olacaktır). Aynı zamanda birçok organizasyonla da uyumsuzdur. OCTAVE-S isteğe bağlı eklenen ihtimallere sahiptir fakat bu OCTAVE'in bir parçası değildir.

OCTAVE birçok çalışma tablosuyla, uygulama pratikleriyle ve 18 cilt içermesiyle geniş ve karmaşıktır.

Web uygulamalarının güvenlik riskleri için deneme listesi sağlamaz

OWASP, uygulamaların tasarımcılarının ve geliştiricilerinin OCTAVE'i kullanmasını beklenmez ve bu yüzden tehdit modellemesinin ana güvenlik noktası kaçmaktadır. Bu ise bir uygulamanın saldırıları sayıf kalması riskini düşürmek için tüm katılımcılar tarafından geliştirme sırasında kullanılmaktadır.

Tehdit modelleme yaklaşımlarının karşılaştırılması

CVSS 'nin STRIDE / DREAD'a kadar kabaca nasıl ölçüldüğü:

Metrik	Öznitelik	Tanım	STRIDE / DREAD'a en yakın
CVSS Taban Metriği	Erişim vektörü	Bölgesel veya uzak erişim?	~ Yararlanılabilirlik
CVSS Taban Metriği	Erişim güçlüğü	Sömürüyü yeniden üretmenin zorluğu	Üretilebilirlik
CVSS Taban Metriği	Belgeleme	İsimsiz veya yetkili?	~ Yararlanılabilirlik
CVSS Taban Metriği	Gizlilik etkisi	Gizliliği ihlal etmenin etkisi	Bilgi Açıklaması
CVSS Taban Metriği	Yoğunluk etkisi	Yoğunluğu ihlal etmenin etkisi	Onaysız değişiklik
CVSS Taban Metriği	Kullanılırlık etkisi	Sistem kullanırlığını ihlal etmenin etkisi	Sistem inkarı
CVSS Taban Metriği	Önyargı etkisi	CIA'a eşit eğilim, ya da CIA'dan bir ya da daha fazlasına eğilim?	Eşitlik yoktur
CVSS Temporal (gecici)	Yararlanılabilirlik	İhmalden yararlanmak ne kadar kolaydır?	Yararlanılabilirlik
CVSS Temporal (gecici)	Düzeltme seviyesi	Sabit bir kullanılabilirlik var mıdır?	Eşitlik yoktur
CVSS Tempora	Rapor güveni	Savunmasızlığın asıl	Eşitlik yoktur

(gecici)		raporu ne kadar güvenilirdir?	
Çevresel CVSS (CVSS Environmental)	İkinci derecede zarar	Eğer tehdit gerçekleşmişse ne kadar kötü zarar verebilir?	Zarar Potansiyeli
Çevresel CVSS (CVSS Environmental)	Hedef dağılımı	Eğer tehdit gerçekleşmişse ne kadar sunucu etkilenir?	Etkilenen kullanıcılar (doğrudan eşit değil)

Alternatif olarak, STRIDE / DREAD CVSS'le nasıl eşleşir:

STRIDE öznitelik	Tanım	CVSS'ye en yakın
Yanıltma kimliği	Kullanıcılar başka kullanıcı olmak için veya başka bir kullanıcı gibi davranabilmek için kontrollerin üstesinden gelebilir?	Doğrudan eşit değil
Veri ile tarif etme	Güvenliği kontrolünü engelleyen uygulamayı almak veya sistemlerin altında yatan temeli (örneğin SQL enjeksiyonu) ele geçirmek için veri saldırgan tarafından kurcalanabilir mi?	Yoğunluk
İtibar	Uygulamanın içinde takip edilebilme eksikliği olduğu için kullanıcılar işlemleri reddedebilir mi?	Doğrudan eşit değil
Bilgi açıklaması	Otoriteler olmaması gereken şeylere maruz kalmış duyarlı bilgilerin yönetilmesini engellemeyi kontrol edebilir mi?	Gizlilik
Servis İnkarı	Saldırgan yetkili bir kullanıcının sisteme erişimini engelleyebilir mi?	Kullanılırlık
Ayrıcalık seviyesi	İsimsiz bir saldırgan kullanıcı olabilir mi veya bir yetkili kullanıcı yönetici gibi davranabilir mi veya aksi halde daha	Doğrudan eşit değil

STRIDE öznitelik	Tanım	CVSS'ye en yakın
	ayrıcalıklı bir rolle değişebilir mi?	

DREAD öznitelik	Tanım … Eğer tehdit gerçekleşmişse	CVSS'ye en yakın
Zarar potansiyeli	Ne zararı meydana gelebilir?	İkinci derecede zarar
Üretilebilirlik	Potansiyel bir saldırı için çalışmak ne kadar kolaydır?	~ Erişim güçlüğü
Yararlanılabilirlik	Bir saldırı çalışması yapmak için neye ihtiyacınız var?(çaba, deneyim)	Yararlanılabilirlik
Ektilenmiş kullanıcılar	Saldırıdan kaç kullanıcı etkilendi?	Hedef dağılımı
Keşfedilebilirlik	Saldırgan tarafından konuyu keşfetmek nasıl kolay?	Doğrudan eşit değil

Genel olarak, CVSS güncel yazılımlar için kullanışlıdır ve gerçekleşen savunmasızlıkların sayısı azdır. CVSS kontrol eden kişiler için benzer risk sıralamasını önemsiz kılmalıdır, fakat birçok önyargı tüm risk hesaplamasını öznel kılar (yerel ve uzak veya hangi bakış açısına göre uygulama daha önemli)ve bu yüzden risk sıralamasının sonuçlandırılmasında anlaşmazlıklar çıkabilir.

STRIDE/DREAD alana yapılan saldırıları azaltmak, tasarım geliştirmek ve savunmasızlıkları ortadan kaldırmadan önce elemek için kullanışlıdır. Aynı zamanda, tehditleri yapısal bir şekilde sayarken ve sıralarken kontrol eden kişiler kullanabilir ve kontrol eden kişilerin benzer önemsiz risk sıralamalarını oluşturur.

Sonuç

Önceki sayfalarda, web uygulamalarının güvenliğinin ana esaslarından bahsettik. Uygulamalar bu kılavuzda bahsedilen özel kontrollerle daha güvenli olacaklar.

Ek Okumalar

- Microsoft, Threat Modeling Web Applications, © 2005 Microsoft http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx?pull=/library/en-us/dnpag2/html/tmwa.asp
- Microsoft, *Threats and Countermeasures*, © 2003 Microsoft
- Howard and LeBlanc, *Writing Secure Code*, 2nd Edition, pp 69 – 124, © 2003 Microsoft Press, ISBN 0-7356-1722-8
- Meier et al, *Improving Web Application Security*: Threats and Countermeasures, © 2003 Microsoft Press
- Saitta, Larcom, and Michael Eddington, *A conceptual model for threat modeling applications,* July 13 2005

  http://dymaxion.org/trike/

  http://dymaxion.org/trike/Trike_v1_Methodology_Document-draft.pdf
- CVSS

  http://www.dhs.gov/interweb/assetlibrary/NIAC_CyberVulnerabilitiesPaper_Feb05.pdf
- OCTAVE

  http://www.cert.org/octave/
- AS/NZS 4360:2004 Risk Management, available from Standards Australia and Standards New Zealand:

  http://shop.standards.co.nz/productdetail.jsp?sku=4360%3A2004%28AS%2FNZS%29

Dökümanın Aslı : OWASP GUIDE 2.0.1

Çeviri : Labris Teknoloji - oguz@labristeknoloji.com (stajyer projesi)