

SAMM (Software Assurance Maturity Model) ile Güvenli Yazılım Geliştirme

Dr. Emin İslam Tatlı (tatli@architectingsecurity.com)
Ocak 2011

1. SAMM nedir?

Yazılım geliştirme süreçleri (Waterfall, Spiral, Agile gibi) temelde planlama, tasarım, kodlama, test, kurulum, bakım gibi benzer basamaklardan oluşurlar. Sonuçta ortaya çıkan sistemin/ürünün güvenli olması isteniyorsa bilgi güvenliği konusu bir süreç gibi algılanmalı ve gerekli aktiviteler yazılım süreçlerinin her basamağına entegre edilmelidir. Bu sayede örneğin planlama aşamasında yazılım geliştiricilere güvenli kod geliştirme üzerine eğitim verilmelisi ya da test aşamasında uygulama seviyesinde otomatik araçlarla güvenlik testleri gerçekleştirilmesi gibi projelerde görmezden gelinen önemli aktivitelerin yazılım sürecine dahil edilmesi sağlanabilir.

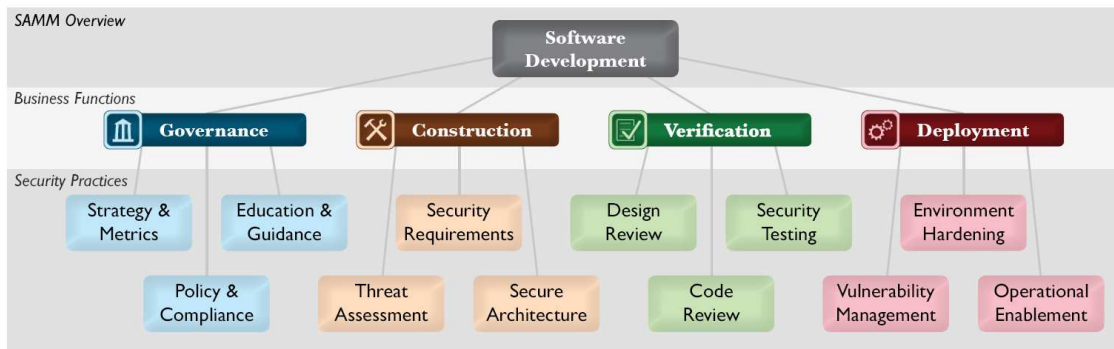
SAMM, bilgi güvenliği aktivitelerinin yazılım geliştirme süreçlerine entegre edilmesini yöneten bir model ortaya koymaktadır. SAMM ilk olarak Fortify firması desteği ile bağımsız güvenlik danışmanı Pravir Chandra tarafından geliştirilmeye başlanmış olup daha sonra OWASP projesine dönüştürülmüştür. SAMM'in başlıca hedefleri şunlardır:

- Organizasyonların yazılım geliştirme süreçlerinde bilgi güvenliği konusunda durumlarını değerlendirmek ve güvenlikle ilgili yapılacak aktiviteleri tanımlamak.
- Dengeli bir yazılım güvenliği sürecini ve her adımda yapılacak güvenlik aktivitelerini belirlemek.
- Yazılım güvenliği sürecini somut bir şekilde desteklemek.

Farklı organizasyonların ve projelerin güvenlik gereksinimleri farklı olabilmektedir. Örneğin, ufak bir ticari proje ile askeri bir projenin güvenlik gereksinimleri, bütçeleri ve de gerçekleştirmeleri gereken güvenlik aktiviteleri farklı olacaktır. SAMM sunduğu modelde bu farkı göz önüne almaktadır. Aynı şekilde, SAMM'in sunduğu model bir organizasyonun tüm yapısı ve projeleri için kullanılabilirliği gibi sadece belirli tek bir projede de kullanılmaya uygundur.

2. SAMM'in yapısı nasıldır?

SAMM'in genel yapısı, Şekil 1'de görüldüğü gibi iş fonksiyonlarından (*business function*) ve bu fonksiyonların gerektirdiği güvenlik aktivitelerinden (*security practice*) oluşmaktadır.



Şekil 1. SAMM'in Genel Yapısı

2.1. SAMM Genel Öğeleri

Bu yapının başlıca öğeleri ve aralarındaki ilişkiler şöyledir:

- En yüksek seviyede başlıca 4 iş fonksiyonu bulunmaktadır. Her organizasyon/proje bu 4 fonksiyonun gereksinimlerini belirli bir seviyeye kadar gerçekleştirmek zorundadır. Bu 4 fonksiyon şunlardır: Yönetim (governance), Yapım (construction), Doğrulama (verification) ve Kurulum (deployment)
- Her iş fonksiyonunun altında toplam 3 adet güvenlik aktivitesi bulunmaktadır. Güvenlik aktiviteleri, ilgili iş fonksiyonunun belirli bir güvenlik olgunluk seviyesine ulaşması için yürütülmesi gereken aktiviteleri temsil etmektedirler.
- Her güvenlik aktivitesi de aynı şekilde 3 adet olgunluk seviyesi (*maturity level*) tanımlanmaktadır ve bir seviyeden diğer bir üst seviyeye geçmek için yapılması gerekenler SAMM tarafından belirlenmektedir.
- Her bir olgunluk seviyesi de, bu seviyeye ulaşılması için yapılması gerekenleri, bu seviyeye ulaşmanın yararlarını, gerekli personel giderlerini ve diğer giderleri, bu seviyeyle alakalı katılımcıları ve diğer ilgili seviyeleri listeler.

2.2. SAMM İş Fonksiyonları ve Güvenlik Aktiviteleri

Aşağıdaki tablolarda dört ana iş fonksiyonu ve bunların güvenlik aktiviteleri detaylı olarak açıklanmıştır:

Yönetim (Governance)	
Yönetim iş fonksiyonu, yazılım geliştirme esnasında yürütülen güvenlik süreçlerinin ve aktivitelerinin nasıl yönetildiği konusu üzerine yoğunlaşır.	
Strateji&Metrikler (<i>Strategy&Metrics</i>)	Bu aktivite gurubu, organizasyon içinde bir yazılım güvenliği yönetim sürecini oluşturmayı, buna bağlı olarak uygulamaların ve bilgilerin risk sınıflandırmasını oluşturmayı ve bu risk sınıflarının güvenlik amaçlarını belirlemeyi sağlar.
Politika&Uyumluluk (<i>Policy&Compliance</i>)	Bu aktivite gurubu, yasal gereksinimleri temel alarak organizasyon içinde güvenlik ve uyumluluk kontrolleri oluşturmayı ve bu sayede yazılım güvenliğini artırmayı hedefler.
Eğitim&Destek (<i>Education&Guidance</i>)	Bu aktivite gurubu, yazılım geliştirme sürecindeki katılımcıların (proje yöneticileri, yazılım mimarları, geliştiriciler gibi) bilgi güvenliği bilinçlerini eğitimler ve teknik destek yoluyla artırmayı hedefler.

Tablo 1. Yönetim İş Fonksiyonu ve İlgili Güvenlik Aktiviteleri

Yapım (Construction)	
Yapım iş fonksiyonu, yazılım geliştirme sürecinde güvenlik amaçlarının belirlenmesi, güvenli yazılım tasarlanması ve geliştirilmesi konuları üzerine yoğunlaşır.	
Tehdit Değerlendirme (<i>Threat Assessment</i>)	Bu aktivite gurubu, yazılımlar için daha etkin risk yönetimi gerçekleştirmeyi, yazılımların karşı karşıya oldukları saldırıları analiz etmeyi ve güvenlik önceliklerini belirlemeyi hedefler.
Güvenlik Gereksinimleri (<i>Security Requirements</i>)	Bu aktivite gurubu, iş hayatındaki başarılı metotları (<i>best practice</i>) dikkate alarak iş akışı esnasında uygulanması gereken güvenlik gereksinimlerini belirler.
Güvenli Mimari (<i>Secure Architecture</i>)	Bu aktivite gurubu, yazılım tasarımı esnasında güvenlik tasarım modellerinden (<i>security design patterns</i>) ve de güvenli mimari ilkelerinden (<i>secure architecture principles</i>) faydalanarak, daha mimari tasarım esnasında güvenliği sisteme entegre etmeyi hedefler.

Tablo 2. Yapım İş Fonksiyonu ve İlgili Güvenlik Aktiviteleri

Doğrulama (Verification)	
Doğrulama iş fonksiyonu, yazılım geliştirme sürecinde ortaya çıkan ürünlerin (mimariyi açıklayan belgeler, kodlar, uygulamalar gibi) doğruluğunu sağlama ve bunları test etme konularına yoğunlaşır.	
Tasarım Denetimi (<i>Design Review</i>)	Bu aktivite gurubu, yapım iş fonksiyonu esnasında oluşturulan yazılım mimarisini güvenlik açısından denetler ve güvenlik gereksinimlerinin tasarlanan mimari tarafından gerçekleştirilip gerçekleştirilmediğini kontrol eder.
Kod Denetimi (<i>Code Review</i>)	Bu aktivite gurubu, yazılım kodunda var olması mümkün güvenlik açıklarını tespit etmeyi hedefler. Bunun için bir kontrol listesi oluşturup manüel denetleme yapılabileceği gibi otomatik araçlar da kullanılabilir.
Güvenlik Testi (<i>Security Testing</i>)	Bu aktivite gurubu, kod denetiminde olduğu gibi yazılımdaki güvenlik açıklarını tespit etmeyi hedefler. Bunun için özel tasarlanmış otomatik araçlar bu amaç için kullanılır. Bu sayede yazılım güvenliğinin belirli bir standarda ulaşması sağlanır.

Tablo 3. Doğrulama İş Fonksiyonu ve İlgili Güvenlik Aktiviteleri

Kurulum (Deployment)	
Kurulum iş fonksiyonu, ortaya çıkan yazılım sürümlerinin nasıl yönetilmesi gerektiği konularına yoğunlaşır. Bu süreç, ürünlerin son kullanıcıya iletilmesi, ürünlerin kurulduğu sunucuların güvenliğinin sağlanması, ortaya çıkan güvenlik açıklarının yamanması gibi aktiviteleri içerir.	
Güvenlik Açığı Yönetimi (<i>Vulnerability Management</i>)	Bu aktivite gurubu, yazılımın kullanımı esnasında ortaya çıkan güvenlik açıklarına karşı gerekli adımları atmayı (açığı inceleme, yama çıkarma gibi) sağlayacak bir açık yönetim süreci oluşturmayı hedefler.
Platform Dayanıklılaştırma (<i>Environement Hardening</i>)	Bu aktivite gurubu, yazılımların kurulu ya da etkileşimde olduğu altyapı bileşenlerinin (işletim sistemi, uygulama sunucusu, veritabanı sunucusu gibi) güvenlik ayarlarının artırılması ve daha dayanıklı hale getirilmelerini hedefler.
İşletim Kurulumu (<i>Operational Enablement</i>)	Bu aktivite gurubu, yazılım geliştiriciler ile operatörler/kullanıcılar arasındaki iletişimi sağlayarak güvenlikle ilgili kritik ayarların kurulum ve kullanım esnasında dikkate alınmalarını sağlar.

Tablo 4. Kurulum İş Fonksiyonu ve İlgili Güvenlik Aktiviteleri

3. SAMM nasıl kullanılır?

Yukarıda SAMM'in genel yapısı hakkında bilgi verdim. SAMM, önerdiği bu modelin pratik olarak projelerde kullanılabilmesi için bir takım araçlar (güvenlik değerlendirme çizelgeleri, skor tablosu kartları, eylem plan şablonları gibi) sunmaktadır. Bu bölümde SAMM'in sunduğu araçları kullanarak SAMM'i projelerimize nasıl entegre edeceğimizi anlatacağım.

3.1. Değerlendirme Çizelgesi (Assessment Worksheet)

Örneğin projeniz esnasında bir takım güvenlik aktiviteleri gerçekleştirdiniz ve SAMM ile güvenlik seviyenizi belirleyip güvenlik açısından durumunuzu görmek istiyorsunuz. Bunun için SAMM'in sunduğu Değerlendirme Çizelgesi (Şekil 2) taslak belgesinden yararlanabilirsiniz.

Değerlendirme Çizelgesinde, her iş fonksiyonu (toplam 4 adet) ve güvenlik aktiviteleri (toplam 12 adet) gruplanmıştır. Her bir güvenlik aktivitesi Evet/Hayır şeklinde cevaplanacak sorular içermektedir. Bu sorular 3 farklı olgunluk seviyesini temsil edecek şekilde gruplanmıştır. Şayet 1. olgunluk

seviyesindeki bütün sorulara Evet şeklinde cevap verebiliyorsanız projeniz o aktivite için birinci seviyeyi başarmış demektir. Aynı şekilde 2. seviyedeki sorular vereceğiniz Evet cevapları sizi ikinci seviyeye ve 3. seviyedeki sorulara vereceğiniz Evet cevapları sizi üçüncü seviyeye ulaştıracaktır. Bunların yanı sıra ara seviyeler de (0+, 1+,2+ ve 3+) vardır. Şayet herhangi bir seviyedeki soruların tamamına Evet demişken bir üst seviyedeki soruların bir kısmını Evet şeklinde cevaplamışsanız bu ara seviyelere ulaşırsınız. Örneğin, birinci seviyedeki sorulara kısmen Evet demişseniz 0+ seviyesine ulaşırsınız ya da üçüncü seviyede ki soruların hepsine Evet şeklinde cevaplayıp bu seviyedeki aktivitelerin de ötesinde aktiviteler gerçekleştirdiyse o zaman 3+ seviyesine ulaşırsınız.

A3 Governance		Assessment Worksheet		
Business Functions	Security Practices	Activities	Answer	Ratings
Governance	Strategy & Metrics	Is there a software security assurance program already in place?	Yes	1+
		Do most of the business stakeholders understand your organization's risk profile?	Yes	
		Is most of your development staff aware of future plans for the assurance program?	Yes	
		Are most of your applications and resources categorized by risk?	Yes	
		Are risk ratings used to tailor the required assurance activities?	No	
		Does most of the organization know about what's required based on risk ratings?	No	
		Is per-project data for cost of assurance activities collected?	No	
		Does your organization regularly compare your security spend with other organizations?	No	
		Do most project stakeholders know their project's compliance status?		
		Are compliance requirements specifically considered by project teams?		
Policy & Compliance		Does the organization utilize a set of policies and standards to control software development?		0
		Are project teams able to request an audit for compliance with policies and standards?		
		Are projects periodically audited to ensure a baseline of compliance with policies and standards?		
		Does the organization systematically use audits to collect and control compliance evidence?		
		Have most developers been given highlevel security awareness training?		
		Does each project team have access to secure development best practices and guidance?		
Education & Guidance		Are most roles in the development process given role-specific training and guidance?		0
		Are most stakeholders able to pull in security coaches for use on projects?		
		Is security-related guidance centrally controlled and consistently distributed throughout the organization?		
		Are most people tested to ensure a baseline skillset for secure development practices?		
		Do most projects in your organization consider and document likely threats?		
		Does your organization understand and document the types of attackers it faces?		
Threat Assessment		Do project teams regularly analyze functional requirements for likely abuses?		0
		Do project teams use a method of rating threats for relative comparison?		
		Are stakeholders aware of relevant threats and ratings?		
		Do project teams specifically consider risk from external software?		
		Are all protection mechanisms and controls captured and mapped back to threats?		
		Do most project teams specify some security requirements during development?		

Şekil 2. Değerlendirme Çizelgesi Örneği

Konunun daha iyi anlaşılması için Eğitim&Destek iş fonksiyonu olgunluk seviyelerinin gereksinimlerine bir göz atalım. Bu iş fonksiyonunun farklı seviyelerine ulaşmak için cevaplanması gereken sorular şöyledir:

Birinci seviyeye ulaşmak için

- Yazılımcıların çoğunluğu, güvenlik konularında farkındalık oluşturan güvenlik eğitimlerine katıldılar mı?
- Her bir proje gurubunun güvenli kod geliştirme destek dokümanlarına erişimi var mı?

İkinci seviyeye ulaşmak için

- Farklı rollere (proje müdürü, yazılım mimarı, geliştirici, test edici gibi) sahip proje katılımcılarının çoğunluğu, rollerine özel güvenlik eğitimine katıldılar mı?
- Proje katılımcılarının güvenlik konularında gerektiği zaman destek alabilecekleri güvenlik uzmanları var mı?

Üçüncü seviyeye ulaşmak için

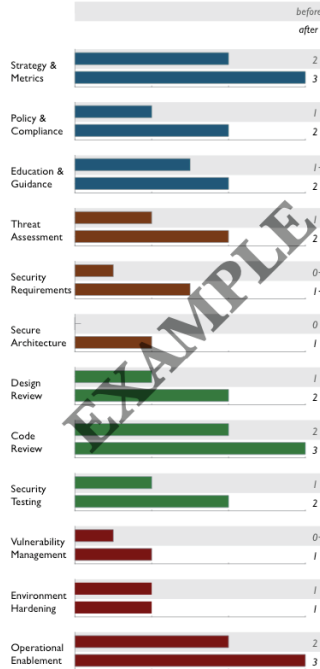
- Güvenlik ile ilgili destekler, merkezi olarak yönetilip bütün organizasyonun bundan yararlanmasına olanak sağlanıyor mu?
- Proje katılımcılarının çoğunluğu, güvenli kod geliştirme konusunda bir sınavdan geçirilerek organizasyon seviyesinde bir temel seviye oluşması sağlandı mı?

sorularına Evet cevabı verilebiliyor olması gerekmektedir.

Değerlendirme sonucu her bir güvenlik aktivitesi için elde ettiğimiz skorları analiz edebilmeli ve bu bilgilerin gelecekte daha güvenli yazılımlar oluşturulmasına katkıda bulunması sağlanmalıdır. Bunun için skor tablolarından ve eylem planlarından yararlanılmalıdır.

3.2. Skor Tablosu (Scorecards)

Değerlendirme çizelgesini kullanarak ölçtüğünüz güvenlik seviyeniz farklı güvenlik aktiviteleri için farklılık gösterebilecektir. Bu seviyeleri kayıt altından tutmak, analiz etmek ve birbirleriyle karşılaştırmak için skor tablolarından yararlanabilirsiniz. Örneğin, farklı zamanlarda gerçekleştirdiğiniz değerlendirmeleri Şekil 3’de gösterildiği gibi skor tablosuna aktarıp önceki ve sonraki seviyeleri gözlemleyebilirsiniz.

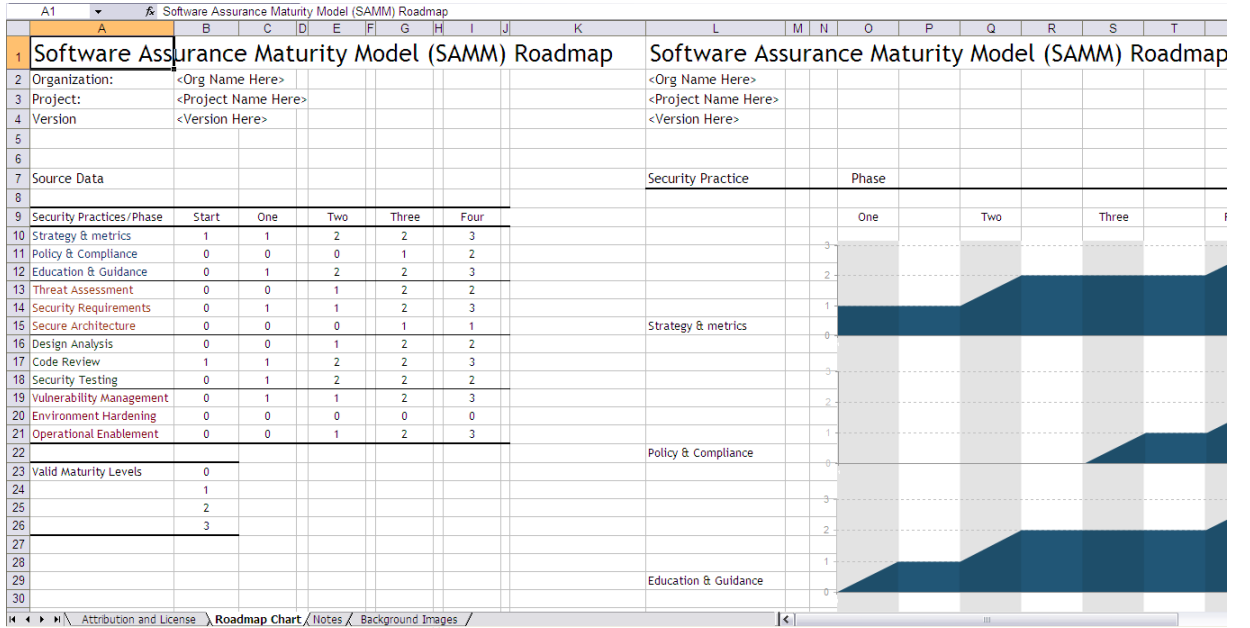


Şekil 3. Güvenlik Değerlendirme Sonucunu Gösteren Örnek bir Skor Tablosu

3.3. Eylem Planı Şablonu (Roadmap Template)

Eylem planı şablonu, yazılım geliştirme sürecindeki güvenlikle ilgili yapılması gereken aktivitelerin ve de gelecekte ulaşılması hedeflenen güvenlik olgunluk seviyelerinin yönetilmesini sağlar.

Bu şablonu kullanmak için öncelikle o anki olgunluk seviyesi değerlendirme çizelgesi yardımı ile belirlenir ve eylem planı şablonuna işlenir. Daha sonra, her bir güvenlik aktivitesi için zamanla ulaşmak istediğiniz seviyeleri belirlersiniz ve bunları da şablona işlersiniz. Böylece bu şablon sayesinde gelecek ile ilgili güvenlik aktivite planlarınızı efektif olarak takip edebilir ve yönetebilirsiniz. Bu şablon her bir güvenlik aktivitesi için hedeflenen olgunluk seviyelerini ayrıca grafiksel olarak da gösterir. Örnek bir eylem planı şablon örneği Şekil 4’te gösterilmektedir.



Şekil 4. Örnek bir Eylem Planı Şablonu

4. Sonuç

Bir OWASP projesi olan SAMM, bir organizasyonun yazılım geliştirme projelerinde güvenlik adına uygulaması gereken aktivitelerini bir süreç olarak belirlenmesine ve yönetmesine olanak sağlar. Bu yazımda SAMM'in genel yapısından ve projelere nasıl entegre edileceğinden bahsettim. SAMM'i daha detaylı olarak öğrenmek isteyenlere referans bölümündeki kaynaklara bakmalarını tavsiye ederim.

5. Referanslar

- SAMM internet sayfası: www.opensamm.org
- SAMM detaylı içeriği : <http://www.opensamm.org/downloads/SAMM-1.0.pdf>
- SAMM Değerlendirme Çizelgesi: <http://www.opensamm.org/downloads/resources/20090925-SAMM-Assessment-v0.4.xls>
- SAMM Eylem Planı Şablonu: <http://www.opensamm.org/downloads/resources/20090610-Samm-roadmap-chart-template.xls>

Yazar Hakkında

Dr. Emin İslam Tatlı, Mannheim üniversitesinde doktorasını tamamladıktan sonra IBM Almanya'da bilgi güvenliği danışmanı olarak çalışmaya başlamıştır. Uygulama güvenliği, Java Enterprise güvenliği, güvenli kod geliştirme süreçleri, SOA mimarileri güvenliği, kimlik ve yetki denetimi, penetrasyon testleri ve uygulamalı kriptografi başlıca çalışma alanlarıdır.

Yazar, www.architectingsecurity.com altında güvenlikle ilgili bir blog tutmakta ve Java Magazin (www.javadergisi.com) dergisi için de güvenlik yazıları yazmaktadır. Kendisine tatli@de.ibm.com email adresinden ulaşabilir ve Twitter/FriendFeed (nickname: eitatli) üzerinden de yazılarını takip edebilirsiniz.