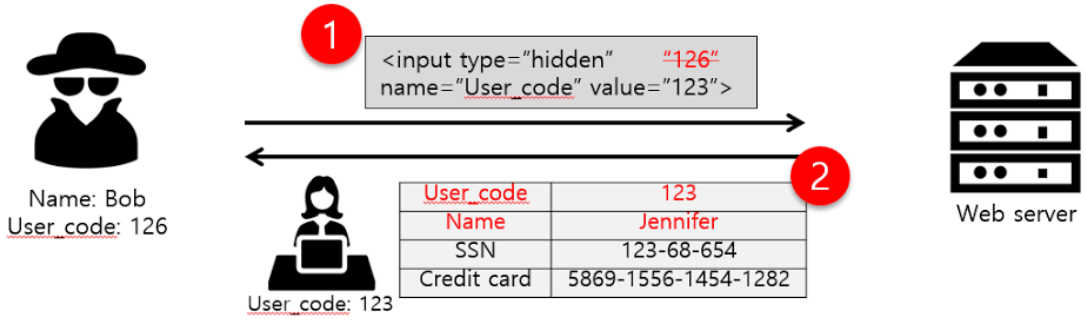
OWASP TOP 10

Herkese selamlar. OWASP TOP 10 zafiyetleri hakkında internetten yaptığım araştırmalardan çıkardığım notları sizlerle paylaşmak istedim. Umarım sizlere faydası dokunur. Keyifli okumalar

1. Broken Access Control

Yetkisiz bir x kullanıcısının yetersiz erişim ayarlarını kullanarak yetkili kullanıcı (mesela Administrator) işlemlerini uygulaması



Görseli açıklamamız gerekirse Bob kullanıcısı 126 id numarası ile istekte bulunuyor ardından isteğini manipüle ederek kendi id numarasını 123 olarak değiştiriyor. Web sunucusu gelen istekte 123 id'sini gördüğü için dönütü de ona göre veriyor. Bu dönüt kullanıcı kritik bilgileri vb. olabilir

Bu zafiyeti önlemek için elimizde çok güzel iki yöntem var istek biçimlerini id değeri gibi değerlere müdahele edilemeyecek şekilde tekrar düzenlemek ve kişiye

ait token kullanmak. Bu yöntemler ile “Broken Access Control” zafiyetinin önemli ölçüde önüne geçmiş oluruz

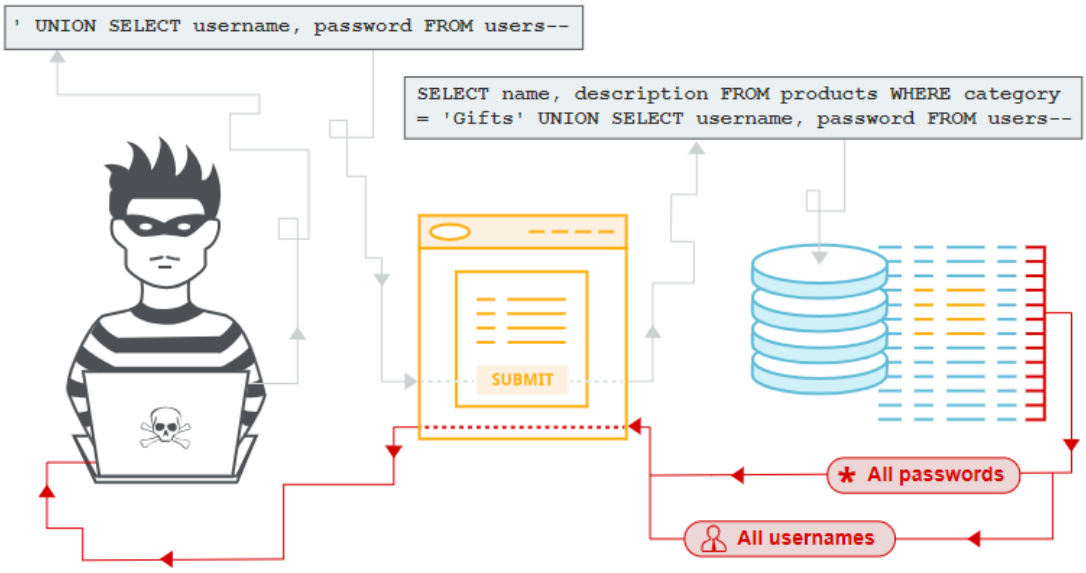
2. Cryptographic Failures

Kullanıcılar açısından önem teşkil eden kullanıcı id, şifre bilgileri gibi verilerin aktarılmasında eksik ya da yanlış şifreleme yöntemleri kullanılarak kritik verilerin kötü niyetli şahısların eline geçmesi durumudur.

Önlenmesi için AES, SHA-256 gibi güvenli algoritmalar tercih edilebilir, halihazırda kullanılan kütüphane ve yazılımların güncellemeleri kontrol edilerek ortaya çıkan açık varsa bunları kapatmaya/önlemeye yönelik adımlar izlenebilir.

3. İnjection

Kullanıcı tarafından girilen verilerin herhangi bir şifreleme ya da filtrelendirmeye tabi tutulmaksızın direk database’ye gönderilip çeşitli parametre veya sorguların işlenmesi ile ortaya çıkan bir zafiyet türüdür



Görseldeki kötü niyetli kullanıcının databaseye gönderdiği sorgu işlemleri herhangi bir filtrelemeden geçmediği için bütün username ve password bilgilerinin şahsın eline geçmesine sebep olur. En basit örneklendirme ile bu şekilde anlatabiliriz.

Engellenmesi için UNION, SELECT, DELETE, “ ‘ ”gibi kelime veya işaretlerin databaseye gitmesini önlemek için yapılacak her hareket ile bu zafiyetin önüne geçilebilir

4. Insecure Design

Bilinçsiz ya da dikkatsiz bir şekilde geliştirilen web uygulamalarında tasarım üzerinde bulunan açıklardır

Parolaların açık bir şekilde saklanması, hata mesajlarının kritik bilgiler içermesi örnek olarak verilebilir.

Önlenmesi için uygulama en başta geliştirilirken her türlü detay için gerekli adımların belirlenmesi gerekir. Plansız bir şekilde yapılan uygulamada açık bulunması muhtemeldir.

5. Security Misconfiguration

Açığın adındanda anlaşılacağı üzere uygulama ya da sistemlerin yanlış konfigüre edilmesi sonucu oluşur.

Önlenmesi için önceki zafiyetteki gibi konfigüre işlemleri için gerekli planlamaların yapılıp ve bu planlamalara uyularak önlenebilir. Aynı zamanda kullanılan yazılımların birbiri ile uyumlu olup gerekli analizlerin yapılması da önleme yöntemlerinden birisidir.

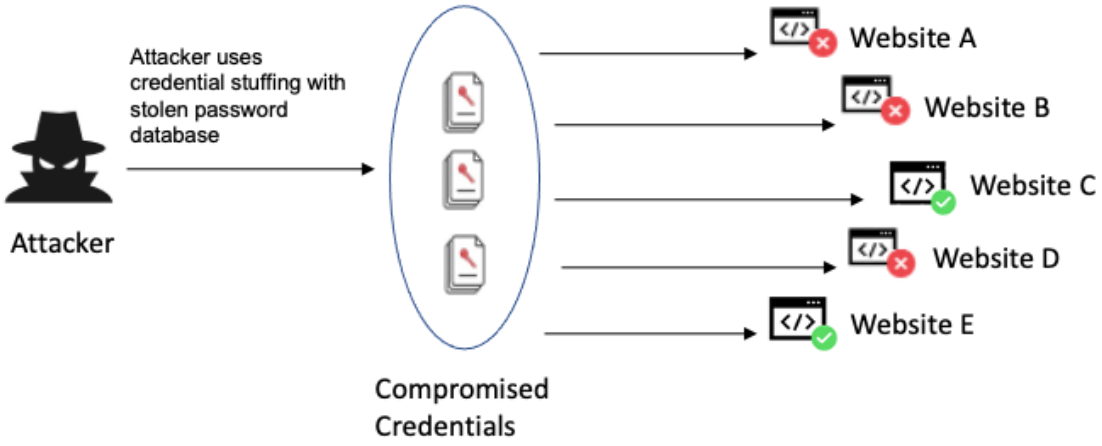
6. Vulnerable and Outdated Components

Web uygulamasının yönetiminde kullanılan kütüphaneler,database yönetim sistemleri, API'lerin kısaca kullanılan 3. parti uygulamaların ortaya çıkardığı zafiyetlerdir.

Önlemek için gereksiz araçlar kaldırılmalı, kullanılan uygulamarın sürümlerinin güncel olduğu düzenli olarak kontrol edilmesi, Kullanılacak uygulamaların kendi kaynaklarından indirilmesi bu zafiyeti önlemek için kullanılacak başlıca yöntemlerden birkaçıdır.

7. Identification and Authentication Failures

Genellikle oturum açma kısmında yapılan hataların ortaya çıkardığı bir zafiyettir. Çok faktörlü kimlik doğrulamasının olmaması, kaba kuvvet saldırılarının engellenmemesi, default username ve parolaların hala kullanımda olması, verilerin clear-text biçimde tutulması gibi sebeplerden ötürü açığa çıkar.



Görseldeki örnekte saldırgan daha önce açığa çıkan verileri elindeki sitelere deneyerek giriş yapmaya çalışır eğer giriş bilgilerini aynı tutan site varsa başarılı bir şekilde giriş yapmış olur.

Engellenmesi için giriş önlemlerinin arttırılması, saniye başına belli bir deneme hakkının olması, yapılan yanlış denemelerinin belli bir seviyeye ulaştıktan sonra deneme yapılmanın engellenmesi, varsa açığa çıkan verilerin değiştirilmesi gibi yöntemlerle zafiyetin sömürülmesi önlenebilir.

8. Software and Data Integrity Failures

Kullanılan uygulamaların güncel olmaması veya yapılan güncellemelerin bilinmeyen kaynaklardan yapılarak zaralı yazılımların uygulamaya entegre olması durumudur.

Önlemek için kullanılan eklenti veya yazılımların güncellikleri düzenli olarak kontrol edilmeli, güncellemelerin güvenilir kaynaklardan yapılması, dijital imza kullanılması zafiyeti önlemek adına izlenilecek adımlardan bazılarıdır.

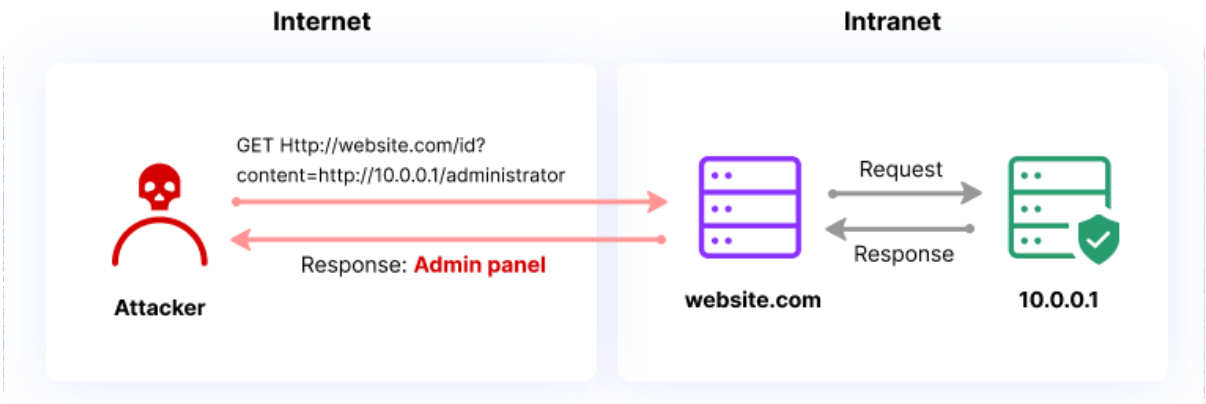
9. Security Logging and Monitoring Failures

Log kayıtlarının düzgün analiz edilmemesi, çeşitli atak vektörlerine karşı kuralların yazılmaması, uygun monitoring araçlarının kullanılmaması ile ortaya çıkan zafiyet türüdür.

Engellenmesi için log kayıtlarının düzenli olarak tutulması ve analizlerinin yapılması. Brute Force gibi saldırı türlerini önlemeye yönelik Yara, Snord, Suricata kurallarının yazılması, daha önce şüpheli işlemler yapan ip'lerin blackliste alınması gibi yöntemler izlenebilir.

10. Server-Side Request Forgery (SSRF)

Saldırganın hedef sunucuya sahte alan adı veya ip üzerinden aradaki doğrulama sürecini atlayıp içeriğe erişmeye çalışmasıdır.



Saldırgan web sitesinin id parametresine özel bi url ekler. Bu url saldırganın erişmek istediği iç ağdaki bir servise yönlendirir. Bu sayede hakkı olmayan yetkileri

kullanarak istediği ağda administrator yetkilerine sahip olup istediği işlemleri uygulayabilir.

Önlenmesi için WAF ürünlerinin kullanılması, trafiğin izlenmesi, yapılacak isteklere kısıtlamalar getirilmesi izlenebilecek adımlardan bazılarıdır.

Bu yazımda sizlere OWASP top 10 zafiyeteri hakkında bilgilrer vermeye çalıştım umarım sizler için faydası olmuştur.

Mansur Derda Şakalar