KARADENİZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
MİKROİŞLEMCİLER LABORATUVARI

Sızma Testi Metodolojisi ve Simülasyonu

1. GİRİŞ

Sızma Testi (Penetration Testing) bir bilgisayar sistemine, ağ sistemine ya da bir uygulamaya saldırgan perspektifinden bakılarak olası zafiyetlerin (vulnerability) tespit edilmesi amacıyla yürütülen bir siber saldırı simülasyonudur.

Günümüzde kişilerin, ticari kurumların ve devletlerin hassas verilerinin dijital kopyalarının siber cihazlarda depolandığı ve işlendiği görülmektedir. Bunun yanında, Nesnelerin İnterneti (IoT) kavramıyla birçok mekanik cihaz siber cihaza evrilmiş ve gündelik hayatımızda verilerimizin canlı olarak her an işlendiği bir döneme girilmiştir.

Bahsedilen durumlar neticesinde bu cihazların (örneğin bilgisayar, veritabanı sunucuları, ağ geçitleri ya da akıllı saatler gibi) var olabilecek güvenlik açıklarının kötü niyetli bir kişiden önce tespit edilip giderilmesi büyük önem teşkil etmektedir.

2. DENEYİN AMACI VE KAZANIMI

Bu deneyde bir sızma testi kapsamında hedef olarak belirlenen cihaza saldırgan bakış açısıyla bir siber saldırı simülasyonu gerçekleştirilecektir. Hedef cihazda çalışan işletim sistemi, uygulama ya da açık protokollere yönelik ne gibi saldırıların nasıl gerçekleştirilebileceği ve bulunan zafiyetlerin nasıl keşfedilebileceği tartışılarak uygulamalı bir şekilde gerçekleştirilecektir. Bu doğrultuda katılımcıların ilgili yetkinlikleri kazanması ve kendi senaryolarında uygulayabilmeleri hedeflenmektedir.

Deney, aynı zamanda katılımcılara aşağıda birkaçı belirtilen ve deney boyunca kullanılacak bileşenler hakkında bilgi edinmelerini ya da mevcut bilgilerini tazelemelerini sağayacaktır:

- Sanallaştırma yazılımı kullanımı ve bir saldırı simülasyonu için gerekli ortamın kurulması,
- Kali Linux dağıtımının incelenmesi, kullanımı ve bu dağıtımda bulunan sızma testi araçlarının kullanımı,
- Windows ve Linux tabanlı makinelerdeki dikkat edilmesi gereken protokoller ve işlevleri**.**
- Keşif, zafiyet taraması ve sömürme (exploit) döngüsünün gösterilmesi üzerine harici erişime açık bir servis kurma.

3. DENEYE HAZIRLIK

Deney süresince en iyi verimin elde edilmesi ve tüm kazanımların edinilebilmesi için deneye gelmeden önce incelenmesi gereken konular aşağıdaki gibidir:

- Sızma testlerinde kullanılan işletim sistemlerinin incelenmesi ve deney boyunca kullanılacak işletim sistemindeki Kali Linux tercihinin araştırılması.
- Sanallaştırma platformlarının (VirtualBox ya da VMware gibi) incelenmesi ve bir ISO kalıbı kullanılarak herhangi bir işletim sisteminin (tercihen Kali Linux) koştuğu sanal makine (Virtual Machine) oluşturulması,
- Harici erişime açık protokollerin en yaygın kullanıldığı bağlantı noktalarının (port) incelenmesi,
- Keşif (Reconnaissance) için kullanılacak nmap aracının incelenmesi ve "-h" komutuyla ya da https://nmap.org/book/man-briefoptions.html adresinden ilgili aracın parametrelerinin incelenmesi.
- Zafiyet, Sömürü, Sömürü-sonrası ve Hak/Yetki Yükseltme (*Vulnerability, Exploit, Post-Exploitation, Privilege Escalation*) gibi kavramlar ve bu kavramlar arasındaki bağlantı incelenmelidir.
- "*cd, ls/dir, mkdir, touch, echo, mv/move*" gibi temel Linux ve Windows komut satırı komutlarına dair bilgiler tazelenmelidir.

4. DENEYİN YAPILIŞI

Deneyde bir sanallaştırma platformu üzerinde saldırgan sanal makine, Windows tabanlı hedef sanal makine ve Linux tabanlı hedef sanal makine şeklinde 3 farklı sanal makine kullanılacaktır. Hedef sanal makineler sızma testi döngüsünün gösterilebilmesi adına kasıtlı olarak çeşitli protokollerinde ya da direkt olarak işletim sisteminde zafiyet bulunduracaktır.

Bunun yanında uygulamalı olarak bazı uygulamaların harici erişime açık bir şekilde kurulup doğru ayarlamalar yapılmadığı ve güncellemelerin gerçekleştirilmediği takdirde nasıl uygulama/servis üzerinden cihazın nasıl sömürülebileceği gösterilecektir.

Sızma testi döngüsü aşağıdaki 6 temel başlıkta incelenecektir:

Ortamın Saldırı Simülasyonu İçin Hazırlanması

En önemli ve ilk adım, saldırı simülasyonu için kullanılacak sanal makinenin sunucu (host) makineden ve ağından izole edildiğinden emin olunmasıdır. Bunun için sanallaştırma platformu ortamının ve sanal makinenin ağ adaptörü ayarlarının nasıl yapılması gerektiği anlatılacaktır.

Kali Linux üzerinde ön bir tanıtım yapılıp seçim sebebi sunulacak, barındırdığı kullanışlı sızma testi araçlarının işlevleri üzerine tartışılacaktır. Ayrıca deneye başlamadan önce ilgili araçların hedef cihazlara ne gibi isteklerde (request) bulundukları ve yanlış kullanımlarının ne gibi suç teşkil edebilecekleri anlatılacaktır.

Hedef Cihaz Üzerine Keşif Gerçekleştirilmesi

İlgili tanıtım, uyarılar ve ortam hazırlıkları tamamlandıktan sonra saldırı simülasyonunun gerçekleştirileceği cihazın keşifi üzerine tartışılacaktır. Bu aşamada "*nmap*" aracından faydalanılacaktır.

Nmap aracıyla:

- Bilinen IP adresi/alan adı üzerine ya da IP bloğu üzerine aktif cihaz tespiti (*Host Discovery*),
- Hedef cihaz üzerinde bağlantı noktası tarama ve çalışan servis tespiti (*Port Scanning & Service Detection*),
- İşletim sistemi tespiti (*OS Fingerprinting*),
- Nmap Scripting Engine (NSE) ile ön zafiyet taraması

gibi işlevler sayesinde hedef cihaz üzerine detaylı bir keşif süreci gerçekleştirilip edinilen bilgilerle bir sonraki adıma geçilecektir.

Servislere Yönelik Zafiyet Araştırması

Hedef cihaza yönelik harici erişime açık ya da filtreli bağlantı noktaları ve bu bağlantı noktalarında çalışan servislerin detaylı bilgileri toplanan bilgiler kullanılarak bir zafiyet araştırması gerçekleştirilecektir. Bu taramada kullanılacak web sitelerinden bazıları:

- Güvenlik araştırmacılarıın birçoğu tarafından kullanılan ve var olan en büyük açık kaynaklı güvenlik açığı ve sömürü (exploit) kodlarına sahip olan "*Exploit-DB"* web sitesi (ya da komut satırı aracılığıyla "*searchsploit"*),
- Resmi CVE veri tabanı olan "*MITRE*" web sitesi,
- Zafiyetin bulunduğu, ancak henüz diğer maddelerdeki platforma yüklenmediği durumlar için en popüler arama motoru Google.

Bu arama servislerinde, ele geçirilen bir "*vsftpd 2.3.4*" ya da "*Apache httpd 2.4.29*" gibi servis bilgileri araştırılacak ve bir zafiyetin varlığı sorgulanacaktır. Bulunduğu takdirde bir sonraki aşamaya geçilecektir.

Metasploit Framework ile Zafiyet Sömürüsü

Metasploit, aktif bir şekilde sürekli olarak güvenlik araştırmacıları tarafından beslenen ve içerisinde birçok hazır sömürü kodu, yardımcı modül ile saldırı sonrası çalıştırılacak kodları (payload) barındıran popüler bir sömürü aracıdır.

Metasploit framework ile sırasıyla:

- Hedef cihazda harici erişime açık çalışan serviste tespit edilmiş ve doğrulanmış olan zayifete uyumlu sömürü kodu bulunur ve seçilir,
- Hedef IP adresi ile hedef bağlantı noktası adresi gibi sömürü kodunun hedefe yönelik çalışması için gerekli ayarlamalar yapılır,
- Sömürü kodu çalıştıktan sonra ne yapılacağını belirleyen bir "*payload*" seçilir,

- Sömürü kodu hedef cihaz üstüne çalıştırılır ve hedef cihazda bir oturum (*session*) elde edilir.

Bu adımlardan sonra artık zafiyetli servis üzerinden hedef cihaza sızılmış ve bir shell oturumu elde edilmiş olur. Bu shell oturumu üzerinden "*sysinfo*", "*ps*", "*getuid*", "*whoami*" gibi sistem içi keşif yapmamızı ve cihaz hakkında bilgi kazanmamızı sağlayacak komutlar çalıştırılabilmektedir.

Sömürü Sonrası Hak/Yetki Yükseltme (Post Exploitation & Privilege Escalation)

Birçok sömürü koduyla sisteme ilk erişim sağlandığında yetki genellikle kısıtlı kullanıcı seviyesinde kalmaktadır. Bu durum, sisteme edinilen erişimdeki yetki seviyesi "*getuid*" komutunun çalıştırılmasıyla anlaşılabilmektedir. Yetki Seviyeleri işletim sistemine bağlı olarak aşağıdaki gibidir:

Windows tabanlı işletim sistemleri için:

- *NT AUTHORITY\SYSTEM*: "*SYSTEM*" yetkisine sahip Windows'taki yetkisi en yüksek hesaptır.
- *<COMPUTER_NAME>\Administrator:* SYSTEM yetkisi kadar olmasa da isminden de anlaşılabileceği üzere erişim edinilen kullanıcı sistem üzerinde muhtemel yönetici bir kullanıcı olyp yüksek bir yetkiye sahiptir.
- *<COMPUTER_NAME>\Ali:* Bu durumda yetkiler USER seviyesinde yani muhtemel olarak en kısıtlı seviyede kalacaktır. Bu noktada başvurulması gereken adım sömürü sonrası hak yükseltme olacaktır.

Linux tabanlı işletim sistemleri için:

- *root (uid=0)*: Bu çıktı Windows'taki SYSTEM yetkisine sahip kullanıcı gibi sistem üzerine tam yetkiyi ifade eder, sistem üzerine elde edilen erişim mutlak kontrol yetkilerine sahiptir.
- *www-data (uid=33)*: Bu çok sık karşılaşılan bir web sunucusuna ait (Apache/Nginx) oluşturulmuş bir hesaptır. Sadece ilgili web sunucusuna ait okuma/yazma gibi sistem üzerine oldukça kısıtlı yetkiye sahiptir. Bu noktada başvurulması gereken adım sömürü sonrası hak yükseltme olacaktır.
- *ubuntu (uid=1000)*: Normal bir kullanıcı hesabı olup sistem üzerine yetkiler yine kısıtlıdır. Başvurulması gereken adım yine sömürü sonrası hak yükseltme olacaktır.

Sömürü sonrası hak/yetki yükseltme işlemleri için yine Metasploit Framework'te bulunan *Meterpreter* payload'larından faydalanılabilmektedir. Meterpreter paylod'u seçildiği takdirde bu payload kodu sömürü koduyla paketlenecek ve o şekilde hedef cihaza iletilecektir. Meterpreter kodunun başarılı olması durumunda ise edinilen yetki seviyesi artmış olacaktır.

Tam Yetki Edinilen Sistem Üzerinde Yapılabilecekler

Tam yetki sağlandıktan sonra sistemde yapılabilecekler aşağıdaki gibidir:

- Kullanıcıya dair hassas bilgilerin okunması, temin edilmesi
- Sistem dosyalarının incelenmesi
- Windows'ta kullanıcıların parola hash'leri SAM (Security Account Manager) adı verilen bir dosyada tutulur ve bu dosya "*C:\Windows\System32\config\SAM*" yolunda bulunur. Bu dosyaya Windows çalışırken SYSTEM yetkilerine sahip olunsa dahi erişilemez. Ancak, Meterpreter'in "*hashdump*" komutu, RAM'de çalışan ve o an oturum açmış kullanıcıların kimlik bilgilerini taşıyan "*lsass.exe"* işlemini hedef alarak ilgili parola hash'lerine sahip olur. Bu hash dosyaları sonrasında Pass the Hash saldırılarında kullanılabilmektedir.
- Hedef makine Linux ise kullanıcı listesinin, ev (*home*) dizinlerinin ve kullandıkları shell bilgisinin tutulduğu "*/etc/passwd*" dosyası okunabilir. Ancak asıl amaç kullanıcı parolalarına dair hash'lerin bulunduğu "*/etc/shadows*" dosyasıdır ve bu çevrimdışı kırılmak üzere bu dosyanın temini hedeflenir.

Sisteme dair parola dosyalarının ele geçirilmesi, çevrimdışı kırılması ya da Pass the Hash ataklarında kullanmasının yanında istenen bir diğer şey kalıcılığın (*Persistence*) sağlanmasıdır. Bu durumda zafiyet düzeltilse dahi yerleştirilmiş bir arka kapı (*Backdoor*) aracılığıyla sisteme her daim erişim sağlanabilecektir.

Temin edilen özet/hash halindeki parola dosyaları ise John the Ripper ya da Hashcat gibi araçlar ve kullanıcıya yönelik hazırlanacak bir sözlük saldırısıyla kırılabilmektedir.