



ÅşalÄ±ÄŸmadÄ±ÄŸÄ±nÄ± anlamasÄ±na olanak tanÄ±r. Qemu gibi popÄ¼ler emÄ¼latÄ¼rler iÄŸin Ä¼zel tespit mekanizmalarÄ± uygulanabilir.

#### Kaynak ve Uygulama:

- Qemu ARM EmÄ¼latÄ¼r Tespiti iÄŸin kullanÄ±lan bir Ä¼rnek: Anti Emulator for Qemu ARM<sup>5</sup>
- EmÄ¼latÄ¼r ortamÄ±nÄ± algÄ±lama ve ÅşalÄ±ÄŸma sÄ¼recinde uygulamanÄ±n iÄŸlevini deÄŸiÄŸtirme.

#### 1.2.0.2 2. Hata AyÄ±klama Modu Tespiti (Debug Mode Detection) Teorik AAŞÄ±klama:

Bir uygulamanÄ±n hata ayÄ±klama (debug) modunda ÅşalÄ±ÄŸmasÄ±, kÄ¼tÄ¼ niyetli kiÄŸilerin uygulamayÄ± analiz etmeleri iÄŸin bir fÄ±rsat saÄŸlar. UygulamanÄ±n hata ayÄ±klama modunda olup olmadÄ±ÄŸÄ±nÄ± tespit etmek, bu modda ÅşalÄ±ÄŸmasÄ±nÄ± engelleyerek gÄ¼venliÄŸi artÄ±rÄ±r.

#### Uygulama Ä¼rnekleri:

1. UygulamanÄ±n ÅşalÄ±ÄŸma zamanÄ±nda hata ayÄ±klama modunda olup olmadÄ±ÄŸÄ±nÄ± kontrol eden kod parÅşacÄ±klarÄ± eklemek.
2. Hata ayÄ±klama modunda olduÄŸunda uygulamanÄ±n ÅşalÄ±ÄŸmasÄ±nÄ± sonlandÄ±rmak veya farklÄ± bir iÄŸlev sergilemesini saÄŸlamak.

#### 1.2.0.3 3. Debugger BaÄŸlantÄ±sÄ± Tespiti (Debugger Attach Detection) Teorik AAŞÄ±klama:

Hata ayÄ±klayÄ±cÄ±larÄ±n (debugger) uygulamaya baÄŸlanması, uygulamanÄ±n izlenmesine ve analiz edilmesine yol aÄŸar. Debugger tespiti, uygulamanÄ±n ÅşalÄ±ÄŸma sÄ±rasÄ±nda bir hata ayÄ±klayÄ±cÄ±ya baÄŸlanÄ±p baÄŸlanmadÄ±ÄŸÄ±nÄ± kontrol eder ve buna gÄ¼re hareket eder.

#### Uygulama Ä¼rnekleri:

1. Debugger tespit edildiÄŸinde uygulamanÄ±n kapanmasÄ±nÄ± veya iÄŸlev deÄŸiÄŸtirmesini saÄŸlama.
2. Hata ayÄ±klayÄ±cÄ±ya baÄŸlantÄ±yÄ± algÄ±layan gÄ¼venlik mekanizmalarÄ± eklemek.

#### 1.2.0.4 4. RootBeer Implementasyonu (RootBeer Implementation) Teorik AAŞÄ±klama:

RootBeer, Android cihazlarÄ±nÄ±n root olup olmadÄ±ÄŸÄ±nÄ± kontrol eden bir kÄ¼tÄ¼phanedir. Root edilmiÄŸ cihazlar, uygulamanÄ±n gÄ¼venliÄŸini tehlikeye atabilir. RootBeer kullanarak, root edilmiÄŸ cihazlarÄ±n tespiti yapÄ±labilir.

#### Uygulama Ä¼rnekleri:

1. RootBeer kullanarak cihazÄ±n root olup olmadÄ±ÄŸÄ±nÄ± tespit etme.
2. Root edilmiÄŸ cihazlarda uygulamanÄ±n ÅşalÄ±ÄŸmasÄ±nÄ± engelleme veya kÄ±sÄ±tlÄ± iÄŸlev saÄŸlama.

#### 1.2.0.5 5. AndroidSecurityManager ile Root Tespiti (AndroidSecurityManager Rooted Device Check) Teorik AAŞÄ±klama:

AndroidSecurityManager, Android cihazlarÄ±nÄ±n gÄ¼venlik durumu hakkÄ±nda bilgi saÄŸlayan bir gÄ¼venlik yÄ¼neticisidir. Root edilmiÄŸ cihazlarÄ± tespit ederek uygulamanÄ±n bu cihazlarda ÅşalÄ±ÄŸmamasÄ±nÄ± saÄŸlar.

#### Uygulama Ä¼rnekleri:

1. AndroidSecurityManager kullanarak root kontrolÄ¼ gerÄŸekleÄŸtirme.
2. Root edilmiÄŸ cihazlarda belirli Ä¼zellikleri devre dÄ±řÄ± bÄ±rakma.

#### 1.2.0.6 6. SafetyNet Implementasyonu (SafetyNet Implementation) Teorik AAŞÄ±klama:

Google SafetyNet, cihazÄ±n gÄ¼venlik durumunu deÄŸerlendirmek iÄŸin kullanÄ±lan bir API'dir. Uygulamalar, SafetyNet ile cihazÄ±n gÄ¼venlik bÄ¼tÄ¼nlÄŸÄ±nÄ± kontrol edebilir ve gÄ¼venlik ihlalleri tespit edildiÄŸinde tepki verebilir.

#### Uygulama Ä¼rnekleri:

<sup>5</sup><https://github.com/strazzere/anti-emulator/blob/master/AntiEmulator/jni/anti.c>

1. SafetyNet API'yi kullanarak cihazın güvenliğini biletinle kontrol etmek.
2. Güvenlik ihlalleri tespit edildiğinde uygulamanın davranışını değiştirmek veya sonlandırmak.

**1.2.0.7 7. Kullanılan Native Kütüphane Checksum Kontrolü (Used Native Library Checksum Control) Teorik Açıklama:** Uygulamanın kullandığı native kütüphanelerin checksum değerlerini kontrol etmek, bu kütüphanelerin değiştirilip değiştirilmediğini anlamamıza sağlar. Bu, uygulamanın güvenliğini korumanın önemli bir yoludur.

**Uygulama Örnekleri:**

1. İşletim sisteminde kullanılan kütüphanelerin checksum değerlerini kontrol etme.
2. Kütüphane üzerinde bir değişiklik tespit edilirse uygulamanın çalışmasını durdurma veya kısıtlama.

**1.2.0.8 8. Tamper Cihaz Tespiti (Tamper Device Detection) Teorik Açıklama:** Cihazın veya uygulamanın manipüle edilip edilmediğini kontrol etmek, uygulamayı güvenli ihlallerine karşı korur. Tamper tespiti ile cihaz veya uygulama üzerinde yapılacak herhangi bir değişikliği algılayabilirsiniz.

**Uygulama Örnekleri:**

1. Cihaz veya uygulamanın tamper edilmiş olup olmadığını tespit etme.
2. Tamper tespit edildiğinde uygulamanın çalışmasını durdurma veya kısıtlama.

**1.2.0.9 9. SSL Pinning ve WebView SSL Pinning (SSL Pinning and Webview SSL Pinning) Teorik Açıklama:** SSL Pinning, uygulamanın belirli bir sunucuya güvenli şekilde bağlanmasını sağlamak için kullanılan bir yöntem. WebView üzerinde SSL pinning uygulamak, kullanıcılara sahte sunucularla bağlantı kurmasını engeller.

**Uygulama Örnekleri:**

1. WebView'da SSL pinning uygulayarak sunucunun kimliğini doğrulamak.
2. Yanlış sunucularla bağlantı kurulduğunda bağlantıyı kesmek.

**1.2.0.10 10. Sunucu Sertifikası Kontrolü (Server Certificate Check) Teorik Açıklama:** Uygulamanın bir sunucuya bağlanırken sunucu sertifikasını doğru şekilde kontrol etmesi, sahte sunucularla bağlantı kurmasını engeller. Bu, man-in-the-middle saldırılarına karşı önemli bir koruma sağlar.

**Uygulama Örnekleri:**

1. Sunucu sertifikasını doğru şekilde doğrulamak.
2. Yanlış sertifika tespit edildiğinde bağlantıyı kesme.

**1.2.0.11 11. Cihaz ve Sürüm Bağlama (DeviceBinding & VersionBinding) Teorik Açıklama:** Cihaz bağlama, uygulamanın belirli bir cihaz üzerinde çalışmasını sağlamak için kullanılan bir yöntem. Sürüm bağlama ise uygulamanın belirli bir sürümde çalışmasını sağlamak için kullanılan bir yöntem.

**Uygulama Örnekleri:**

1. Uygulamanın sadece belirli bir cihazda çalışmasını sağlamak için cihaz bağlama işlemi gerçekleştirme.
2. Uygulamanın yalnızca belirli sürümlerde çalışmasını sağlamak için sürüm bağlama işlemi.

**1.2.0.12 12. T  ketici Do  rulas   (Consumer Verification) Teorik A    klama:**  
Uygulaman  n ger   ek kullan  c   taraf  ndan kullan  ld      n   do  rulamak, sahte kullan  c  lar   ve otomatik i  lemleri engellemeye yard  mc   olur. Bu do  rulama i  lemi, t  keticinin kimli  ini do  rular.

**Uygulama   rne  leri:**

1. T  ketici do  rulas   i  sin g  venlik testleri ve algoritmalar kullanmak.
2. Do  rulanmam     kullan  c  lar i  sin eri  im k  s  tlamalar   koymak.

*6.Hafta – Sonu*