
GAZİ ÜNİVERSİTESİ
Gazi
1926

ANDROID SISTEMLERDEN
DELIL AMAÇLI IZ TESPİT ETME YÖNTEMLERİ

Özgür KOCA

MAKALE ÇALIŞMASI

ADLİ BİLİŞİM ANABİLİM DALI

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

MAYIS 2017

ÖZET

Cep telefonları, özellikle de akıllı telefonlar hayatımızda önemli bir rol oynamaktadır. Mobil cihaz pazarının muazzam büyümesiyle, onları suç faaliyetinde kullanma imkânı da sürekli artmaktadır. Android, piyasadaki son derece rekabetçi platformlardan biridir. Birçok üretici tarafından kullanılan Android, farklı cihaz modellerini çalıştırmak için kullanılmakta ve bu da güçlü bir çeşitliliğe neden olmaktadır. Böylece, Android tabanlı akıllı telefonların fiziksel imaj ediniminin zorluğu, özellikle son Android sürümünün kaynak kodunun çok geç yayınlanması ile daha da anlam kazanıyor. Sonuç olarak, en yeni sürüm işletim sistemine sahip mevcut akıllı telefonların, mevcut akıllı telefon adli araçları kullanılarak edinilmesi de güçleşiyor. Bu çalışmada, mantıksal ve fiziksel olarak edinme olanağı sunan (imaj alma) adli mobil cihaz araçlarının kapsamlı bir perspektifi verilmektedir. Ayrıca yazılım araçları kullanılarak gerçekleştirilen edinimlerin sınırlılıklarını aşmak için iki yeni yöntem de incelenmiştir. Birinci yöntem firmware güncelleme protokolünün zafiyetleri kullanılarak fiziksel imaj elde etmeyi incelerken, ikinci yöntemde anti-adli bilişim tedbirleri uygulanmış bir Android mobil cihazdan delil elde etmek için cihaz yönetim yazılımı Droidjack araştırılmıştır. Son olaradak da adli araştırmacının hedefinde olan veri setlerini / delil izlerini üreten Android uygulamaların programlama teknikleri açısında sunduğu veri mahremiyetleri incelenerek bir anti-adli bilişim perspektifi ortaya koyulmuştur.

Anahtar kelimeler: *physical image; physical acquisition; Android forensic; forensic tools, Android adli bilişim, Android fiziksel edinim, Firmware (bellenim) güncelleme protokolü, Flash bellek okuma komutu, Bootloader, Android Fastboot, JTAG, dowload mode*

İçindekiler Tablosu

1. GİRİŞ

Mobil cihazların kullanımı, özellikle akıllı telefonların kullanımındaki artış ile dijital suçlar da arttı. Akıllı telefonların ortaya çıkışı, insanların yaşama, çalışma ve oyun yapma biçimini tamamen değiştirmiştir. Bununla birlikte, suç işlemede akıllı telefon kullanım oranının artması ile adli araştırmacıların kanıt elde etmek için zanlıların akıllı telefonlarını mahkemelerde kullanımları da artmıştır. Akıllı telefonlardan elde edilen kanıtlar, mahkeme salonunda diğer kanıt şekillerinden farklı olarak, kabul edilebilmeleri için güvenilir olmalıdır.

Son birkaç yılda Android akıllı telefonları hedef alan önemli miktarda bellek edinimi araştırması yapıldı. Edinme amacı, silinen veriler de dahil olmak üzere yararlı bilgiler toplamak ve daha fazla analiz etmek ve mahkemeye sunmaktır. Adli mobil cihazlar için iki temel edinim yöntemi vardır: fiziksel ve mantıksal. Fiziksel edinme, silinen veriler de dahil olmak üzere tüm fiziksel depolama alanının bit kopyasıdır. Mantıksal edinim, bir dosya sisteminin bir parçası gibi mantıksal depolamayı elde eder. Yani elde edilen veriler edinim yapılan sistemin dosya sistemi tablosunun sağladıklarıdır. Akıllı telefonlarda saklanan veriler kırılgan olabilir, çünkü veriler üzerine yazılabilir veya silinebilir. Bu nedenle, silinen verileri elde etmek ve ayıklamak için mantıksal edinim yerine fiziksel edinim kullanma ihtiyacı vardır [1].

Fiziksel edinim araçları, sabitleştirilmiş ve yazılım tabanlı araçlara sınıflandırılmıştır. Donanım tabanlı yöntem, işletim sistemini fiziksel bir aygıtla bypass etmektir. Böylece hedef sistemin dosya yerleşim tablosunun sağlamadığı veriler gibi dez avantajlar ortadan kalkar. Özel bir iletişim portu, dahili belleği kopyalamak için özel bir donanımla açılır [2]. Android akıllı telefonlarda, JTAG test pinleri bir cihazın dahili belleğini almak için kullanılabilir [3]. Bununla birlikte, tüm Android akıllı telefonlarda JTAG test pinleri bulunmamaktadır. Yazılım tabanlı yöntem, dahili belleği elde etmek için hedef işletim sistemi üzerinde çalışan bir yazılım kullanmayı gerektirir [2]. Android akıllı telefonlarda bir seçenek de **/dev/mem** aygıtlarından veri edinmektir [3]. Maalesef bu yöntem yalnızca en fazla 896 MB RAM'li akıllı telefonlar için geçerlidir [4]. *Kollar* [5], **fmem** adında fiziksel

edinim için /dev/mem aygıtını kullanan yüklenebilir bir çekirdek modülü geliştirdi. Ancak, bu modül tüm Android akıllı telefonlar için geçerli değildir [4].

Android akıllı telefonlardan fiziksel olarak veri edinmek için, genellikle akıllı telefonun, özel önyükleyici, özel kurtarma modu veya kök erişimi olan normal modda [6] önyüklemesi yapılmalıdır. Ardından, donanım cihazına veya sunucusuna (ör. Dizüstü veya masaüstü) imaj verisi göndermek için akıllı telefonda ilgili kodu çalıştırılır [6].

Akıllı telefon işlemci hızları, kullanılan kablo türleri ve aktarılan veri miktarı nedeniyle fiziksel olarak edinme süreci zaman alıcı olabilir. Bazen fiziki edinimin tamamlanması saatler alır. UFED ve Oxygen Forensic gibi ticari araçların çoğu USB üzerinden veri gönderir. Bununla birlikte, kopyalanan verilerin iletim hızı, USB'nin maksimum iletim hızını kullanmaz. Örnek vermek gerekirse, USB 2.0, maksimum 480 Mbps iletim hızına sahiptir, ancak en fazla 320 Mbps alır [7]. 2016'da piyasadaki en büyük Android akıllı telefonlar 128 GB'tır. Akıllı telefonlar büyümeye devam ederken, fiziksel olarak onları edinim süreleri de artacaktır [6].

Bu makalede, Android akıllı telefonlar için birçok farklı fiziksel edinim aracını analiz ettik ve maliyetleri, bütünlüğü, veri kurtarma, kullanışlılık, adli veri aşamalarını dışa aktarma yolları ve genel Android akıllı telefonu destekleme yöntemlerini karşılaştırdık.

2. ANDROID'İN MİMARİSİ

Android'in iç tasarımını ve mimarisini anlamak, Android'in esnekliğinden dolayı adli bir soruşturmada en önemli konulardan biridir. Android platformu, yeni sürümlerle zaman içinde değişiyor. Sürümler arasındaki farklılıklara göre, mimari de farklılaşmaktadır. Bununla birlikte, Android mimarisinin ana çekirdek bileşenleri aynıdır. Android mimarisi, Şekil-1'de gösterildiği gibi dört ana katmandan oluşur:

2.1. Linux Çekirdeği

Android çekirdeğini anlamak en önemli unsurdur, çünkü Android mimarisinin temelini oluşturmaktadır [8]. Bellek, ağ ve süreç yönetimi ve güvenlik gibi temel hizmetleri destekler. Ayrıca neredeyse tüm donanım için çeşitli sürücüler de barındırır [8, 9].

2.2. Kütüphane ve Android Runtime

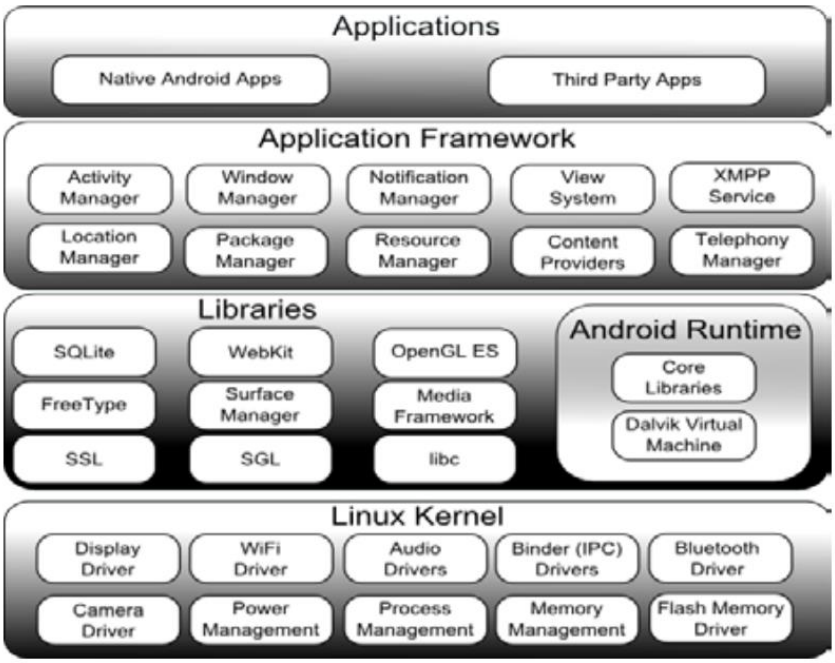
Android, C/C++ [8] ile yazılmış kütüphaneler seti içerir. Standart CSystem Kütüphanesi, Medya Kütüphaneleri, 3D Kütüphaneler gibi kütüphaneler, sistem bileşenleri tarafından Uygulama Çerçevesi katmanı [9] vasıtasıyla kullanılır. Android çalışma zamanı (runtime) bölümü, Android için özel olarak tasarlanmış ve optimize edilmiş bir tür Java Sanal Makinesi olan Dalvik Sanal Makinesi (DVM) adı verilen önemli bir bileşen sunmaktadır [8]. Ayrıca, geliştiricilerin standart Java programlama dili [8] kullanarak Android uygulamaları yazabilmesini sağlayan çekirdek kütüphaneleri seti de sağlar. Çekirdek kütüphanelerin ve DVM'nin bir kümesi, çalışan her uygulamanın DVM'nin kendi örneğini bulundurduğu ve kendi işlemi içinde yürüdüğü bir Android çalışma zamanı oluşturur [9].

2.3. Uygulama Çerçevesi

Bu katman, Java uygulamalarına istismar edilebilecek birçok üst düzey hizmet sunmaktadır [8, 9]. Uygulama geliştiricileri, Çerçeve tarafından uygulanan güvenlik kısıtlamalarına her zaman saygı duyan geniş bir Uygulama Programlama Arabirimi (API) seti aracılığıyla hizmet sunabilir ve bunları sağlayabilirler [9].

2.4. Uygulama

En üst katman, Java Programlama Dili [10] ile yazılmış bir program demetini (ör. Iletişim yöneticisi, takvim, SMS programı, web tarayıcısı, bir e-posta istemcisi) içerir.



Şekil 1 - Android Mimarisi

3. VERİ EDİNİM PROSEDÜRLERİ

Adli incelemecilerin benimseyebileceği, Android akıllı telefonlardan veri toplama sürecinin farklı uygulama senaryoları vardır. Uygun prosedürü kullanarak, adli bilişim uzmanları akıllı telefondan maksimum bilgiyi alabilir, böylece elde edilen verilerin mümkün olduğunca daha güvenli ve en az müdahaleci bir şekilde analiz edip belgelendirilebilir. Adli inceleyici, hedef akıllı telefonda kayıtlı verileri korumak için gerekli prosedürleri izlemelidir [11].

3.1. Erişim Kontrolü Prosedürü Olmadan Akıllı Telefonun Veri Edinimi

En basit olan bir durum, çıkarılabilir hafıza kartı ile kilitli olmayan bir akıllı telefonun edinimidir. Daha önce de belirtildiği gibi, inceleyici ilk önce hafıza kartlarından veri çıkardıktan sonra kopyaları alınan adli inceleme kartlarını akıllı telefona tekrar takmalıdır. Ardından inceleyici Android akıllı telefonda süper kullanıcı ayrıcalıklarının durumunu kontrol etmelidir (super su, root). Etkinleştirilirse, inceleyici USB hata ayıklama aracı ADB'yi kullanarak dahili belleğinin bir kopyasını oluşturarak kısıtlama olmaksızın akıllı telefondaki depolanmış verilere erişebilir. Ancak, akıllı telefonda süper kullanıcı ayrıcalıkları devre dışı bırakılırsa, bu durumda bazı Android akıllı telefonlar bootloader modu veya kurtarma modu kullanılarak edinilebilir. İnceleyici, bu teknikleri bu tür akıllı telefonlara uygulama imkânını değerlendirmelidir. İncelemeciler tarafından kullanılabilen mevcut mobil cihaz adli araçları, Cellebrit UFED ve Oxygen Forensic gibi verileri edinmek için kullanıcı ayrıcalıklarını kullanmazlar. Bunun yerine Cellebrit UFED, bootloader modunu kullanır. Dahili belleğin tam bir kopyasını kurtarmak için etkili bir mobil aygıt adli aracı seçmek, adli bilişim görevlisine kalmıştır.

3.2. Erişim Kontrol Prosedürüyle Akıllı Telefonun Veri Edinimi

Android çalışma zamanı bölümü, Android için özel olarak tasarlanmış ve optimize edilmiş bir Java Sanal Makinesi türü olan Dalvik Sanal Makinesi (DVM) adı verilen önemli bir bileşen sunmaktadır [8]. Ayrıca, geliştiricilerin standart Java programlama dili [8] kullanarak Android uygulamaları yazabilmesini sağlayan çekirdek kütüphaneleri seti de

sağlıyor. Çekirdek kütüphanelerin ve DVM'nin bir kümesi, çalışan her uygulamanın DVM'nin kendi örneğini bulundurduğu ve kendi işlemi içinde yürüdüğü bir Android çalışma zamanı oluşturur [9]. Android akıllı telefon, bir şifre veya desen gibi erişim kontrolü kullanarak kilitlenebilir. NIST'e göre [12], kilitli akıllı telefonlara erişmenin üç yolu vardır:

1. Araştırmacının olası geçerli parolaları istediği araştırma yöntemi.
2. Araştırmacının akıllı telefona erişmek için yıkıcı olmayan bir prosedürü gerçekleştirmesi gereken donanım yoluyla erişim. Bu yöntem üreticilerin ve yetkili servis merkezlerinin desteğini gerektirmektedir.
3. Yazılımsal erişim yöntemleri, mobil cihaz modeline ve Android sürümüne bağlı olsa da genellikle en kolay yoldur.

İnceleyici kanıttan ödün vermekten kaçınmak için en az müdahaleci yöntemi kullanmalıdır. Akıllı telefon ele geçirildiğinde şifre veya model elde edilmişse, test edilmelidir. İnceleme başarılı olmazsa, akıllı telefonun bir ADB aracı kullanarak USB hata ayıklama bağlantılarını kabul edecek şekilde yapılandırılıp yapılandırılmadığını kontrol etmelidir. Başarılı olursa, edinme sürecini devam ettirmek için süper kullanıcı erişim denetimi ayrıcalıkları kazanmaya çalışmaktadır. Akıllı telefona süper kullanıcı erişim kontrolü için herhangi bir ayrıcalık olmadığında bile, denetçi, erişim kontrol sistemini atlamak için ADB aracı aracılığıyla uygulamalar yükleyebilir. Erişim kontrol sistemini atlamak mümkün olmadığı veya USB hata ayıklama erişiminin devre dışı bırakıldığı durumlarda, akıllı telefona takılabilen çıkarılabilir hafıza kartından veriler alınabilir.

4. FİZİKSEL EDİNİM

Daha önce bahsedildiği üzere edinimi gerçekleştirilecek cihazın kendi işletim sistemine ait dosya sistemi içerik sağlayıcısı aracılığı ile elde edilen edinimler mantıksal edinim olarak adlandırılır. Bu yöntemle edinilecek veriler disk ortamında bulunan verilerin birebir karşıklıkları değil işletim sisteminin sunduklarıdır. Böyle bir edinimde doğal olarak kullanıcı tarafından silinmiş dosyalar edinilemez.

Fiziksel edinim ise verilerin kayıt edildiği disk ortamına (RAM, eEPROM, ROM vb) işletim sistemi fonksiyonlarını devreden çıkartarak doğrudan erişmeyi ifade eder. Bu yöntemde dosya sistemi bilgileri kullanılsa da elde edilebilecek veri miktarı çok daha fazladır ve sistem hakkında birçok bilgi sunar. Fiziksel edinimde edinimi gerçekleştiren araç doğrudan disk verisini kendisi okur.

4.1. Linux Bellek Çıkarıcı (LiME)

LiME aracı 2012'de J. Sylve ve diğerleri tarafından piyasaya sürülmüştür. [4]. Android'den, uçucu belleği elde (RAM) etmek için kullanılan açık kaynaklı bir adli araçtır. LiME, bellek sayfalarını adli olarak sağlam bir şekilde elde edebilen **dmd** adlı yeni bir yüklenebilir çekirdek modülünü temel almaktadır. Akıllı telefondaki SD'ye veya ağ üzerinden hafızayı almayı destekler. Dmd modülü şööyle gibi çalışır: sistem RAM'inin fiziksel bellek adres aralıklarını öğrenmek, her bellek sayfasında fiziksel adresleri sanal adreslere çevirmek, tüm bellek sayfalarını okumak ve TCP soketinin SD'sine yazmak için çekirdek yapısını ayrıştırır. LiME aracı önemli özellikler sunar [13]: edinme yani hedef cihaza aktarmak için sadece dmd modülü gereklidir, bellek dökümü için çok az sayıda çekirdek işlevi gereklidir, dmd modülünün yüklenmesi asgari ayak izi oluşturur ve kullanıcı alanı ile minimum etkileşimi kurar. Sonuçlar sayfaların yaklaşık %99.46'sının TCP bağlantısı üzerinden doğru olarak yakalandığını ve sayfaların % 99.15'i SD karta doğru yazıldığını gösteriyor. Önerilen modül tüm Android cihazlarını destekliyor ancak yine de genel bir modül olarak değerlendirilmiyor. Ayrıca, Wächter [14], modelin belirlenmesi, Android sürümünün belirlenmesi, ekranı kilitleme, süper kullanıcı yetki istismarı, kaynakların kullanılabilirliği,

çekirdek konfigürasyonu ve kanıt erozyonu nedeniyle LiME aracının kollukta adli olmasının pek çok nedenden dolayı mümkün olmadığı sonucuna varmıştır.

4.2. Android Fiziksel Döküm (APD)

APD, S. Yang ve ark. Tarafından geliştirilmiştir. [15]. Bu araç, Android akıllı telefonların bellenim güncelleme protokollerini analiz etmeye dayanmaktadır. Bu nedenle, Android cihazların önyükleme yükleyicisinin Android **güncelleme protokolleri** aracılığıyla dahili belleğe erişir. Hem bölüm hem de tüm belleğin dökümünü almayı destekliyor. APD'yi kullanarak edinilen verilerin biçimi, akıllı telefon adli analiz araçları aracılığıyla analiz edilebilen ham verilerdir. Yazarlar, önerilen yöntemin edinilen verilerin bütünlüğünü garanti ettiğini ispatladı. APD'nin veriyi yüksek hızda aldığını gösterdiler; UFED 4PC ortalama 120 dakika sürerken, 32 GB belleğin dökümünü almak yaklaşık 30 dakika aldı. APD, ekran kilidi nedeniyle kısıtlamaya rağmen yürütülebilir; Normal önyükleme modundan ziyade telefonu kapatarak ve bellenim güncelleme modunda yeniden başlatarak. APD aracı, en yeni Android modellerinin 80'inden fazlasını destekler. Bununla birlikte, yöntemin en büyük dezavantajı, yeni Android akıllı telefonlar her başlatıldığında üretici yazılımı güncelleme protokolünü analiz etmeyi gerektirmesidir.

4.2.1. Hawkeye

Hawkeye, Guido ve ark. Tarafından 2016 yılında fiziksel edinim amaçları için önerildi. [6]. Hawkeye'nin amacı, fiziksel edinim sırasında aktarılması gereken veri miktarını ve gereksiz verileri azaltmaya odaklanıyor. Böylece, toplam edinim süresini azaltır. Hawkeye, Android akıllı telefonları fiziksel döküm elde etmek için özel açılış veya kurtarma kipinde çalıştırıyor. Araç, tescilli marka aracını geçici olarak hedef akıllı telefonun RAM belleğine yükler. Araç temel hash ve bölümlerin bir listesini de sağlar. Araç, daha sonra USB aracılığıyla arka uç PMF mimarisine gerekli veri bloklarını belirleyip gönderecektir. Yazarlar birkaç nedenden ötürü PMF'yi seçtiler: resimleri otomatik olarak geri yazarak ham formata çevirme. Araç, bir bölümü veya akıllı telefonun dahili belleğini tam olarak alabilir. Hawkeye 16GB boyutunda dahili belleği 7 dk içinde başarıyla elde edebildi.

4.2.2. Adroid Memory Extractor (AMExtractor)

AMExtractor [3], Android cihazlardan uçucu hafıza (RAM) elde etmek için kullanılan bir araçtır. AMExtractor çekirdek alanında kod yürütmek için /dev/kmem aygıtını kullanır. Bu, yüklenebilir çekirdek modülünün kısıtlamasını önleyecek ve herhangi bir değişiklik yapmadan en son stok ROM'larda çalışma olanağı sağlayacaktır. AMExtractor hedef akıllı telefonun kaynak koduna ihtiyaç duymaz ve çoğu Android işletim sistemi sürümüyle uyumludur. Diğer araçların aksine AMExtractor, hedef akıllı telefonlar üzerinde minimum etkiye sahip olduğu için çekirdek modunda çalışır. Ayrıca, cihazı çekirdek modunda çalıştırmak, veri kopyalamayı en aza indirir ve gizli verileri kullanıcı modunda iken inceler. H. Yang ve ark. [3], AMExtractor kullanılarak elde edilen verilerin, LiME'yi kullanarak elde edilen verilerle neredeyse aynı olduğunu gösterdi.

4.2.3. Androphsy

ANDROPHSY, 2015 yılında I. Akarawita ve ark. tarafından geliştirilen açık kaynaklı bir araçtır. [16]. Dijital adli süreçlerin tüm safhalarını destekleyen ilk açık kaynak araçtır. ANDROPHSY mimarisi dört ana modülden oluşur: vaka işleme, edinme, analiz desteği ve raporlama modülü. Vaka işleme modülünde, özel durum için vaka oluşturma ve yedek arşiv işlevleri sağlanmaktadır. Edinim modülü fiziksel ve mantıksal edinim sağlar. Analiz modülünde, çıkarılan verilerin tam bir inceleme ve analizi. Ve son olarak raporlama modülü, PDF formatında bir rapor oluşturmayı sağlar. Bu aracı kullanmak için tek bir .jar dosyası ve yapılandırma komut dosyası ayrı ayrı kurulmalıdır. Bu araç ile fiziksel edinime odaklanan yazarlar, dd ve Android Debug Bridge (adb) komutları gibi düşük seviyeli Linux ve Android dahili adli işlevlerini ağırlıklı olarak kullanmışlardır. Adb komutları, Android akıllı telefonu ve USB üzerinden bağlı iş istasyonu arasındaki bağlantı ve iletişim süreçlerini yönetmek ve gerçekleştirmek için kullanılır. Dd komutu, ham görüntüleri fiziksel sürücülerden kurtarmak için kullanılan yerleşik bir komut satırı yardımcı programıdır. ANDROPHSY, veri değişimini en aza indirgemek için Linux çekirdeğini kök erişimi kazanmak için kullanır. Kimlik doğrulama ve gizlilik için kullanıcı erişim kontrolü ve vaka yönetimi sağlar. SD kartı edinim hedefi olarak kullanmaz. Bunun yerine, veriler TCP bağlantısı üzerinden aktarılır.

4.2.4. Android Digital Autopsy (ADA)

ADA, 2016 yılında R. Fasra ve diğerleri tarafından geliştirilen, açık kaynak kodlu bir dijital adli araçtır. [17]. Araç, fiziksel, mantıksal ve dosya sistemi edinimi gerçekleştirir. Yazarlar multimedya kartı (MMC) bölüm düzenine sahip bir cihaz kullandılar. Fiziksel edinme sürecini otomatikleştirmek için geliştirilmiş bir komut dosyası yazıldı. Komut dosyası veri bloklarını tanımlar, daha sonra kök erişimini kazandıktan sonra blokların RAW görüntülerini elde etmek için dd komutunu kullanır. Elde edilen veriler, daha sonra harici bir hafıza kartına, yani SD karta depolanmaktadır. Önerilen araçla birlikte, yazarlar ADA Analiz Aracı'nı geliştirdiler, ancak ne yazık ki bu mantıksal edinim içindir.

4.2.5. Cellebrite UFED

Cellebrite UFED [18], Android gibi çeşitli cihazlar ve platformlarda fiziksel, mantıksal, dosya sistemi ve şifre alımını gerçekleştiren ticari bir adli araçtır. Ayrıca, çözme, analiz ve raporlama da gerçekleştirir. UFED, tüm Android işletim sistemi sürümlerinden veri edinebilir.

4.2.6. Oxygen Forensic Suite

Oksijen Forensic Suite [19], çok çeşitli akıllı telefonları destekleyen önde gelen adli tıp araçlarından biridir. Bu, tüm dünyada 50'den fazla ülkede Yasa Uygulayıcılar, ordu, polis departmanlrı ve diğer hükümet yetkilileri tarafından kullanılır. Araştırmacılar, Android akıllı telefonlarının fiziksel olarak edinilmesini, gelişmiş incelemesini ve akıllı telefondan çıkarılan ham görüntülerin ve cihaz görüntülerinin analizini yapmalarını sağlar. Desteklenen akıllı telefonların tamamen otomatik bir şekilde edinilmesi ve analiz edilmesine olanak tanır. Yaklaşık 45 dakika içinde 16 GB akıllı telefon edinebilir. Akıllı telefon faaliyetlerini özetleyen denetmen için iyi tanımlanmış bir rapor sunar.

4.2.7. XRY Physical

MSAB [20], özütleme, analiz etme ve raporlama için ürünler sağlar. XRY Fiziksel araç, hedef cihazı değiştirmeden dahili belleğin ve çıkarılabilir medyanın çıkarılmasını destekler. Ayrıca, kullanıcıların bellek imajının karma değerlerini ve ayrıca tek tek çözülen dosyaları oluşturmalarına olanak tanır. XRY Fiziksel, işletim sistemini atlayarak hedef akıllı telefondan ham verileri kurtarır ve silinen verileri hedef akıllı telefondan daha derinlemesine gidip kurtarma şansı sunar. Fiziksel özütleme iki ayrı aşamaya ayrılır: ilk veri tabanı, ham veriler akıllı telefondan alınır ve kod çözme aşaması, burada araç veriyi otomatik olarak anlamlı bilgiler haline getirir. Çıkarılan veriler XAMN Spotlight tarafından görüntülenebilir.

4.2.8. Device Seizure

DS [21] fiziksel, mantıksal, dosya sistemi ve şifre elde etmeyi desteklemektedir. Edinilen tüm veriler hakkında eksiksiz bir analiz ve rapor sunar. Geniş platform ve cihaz yelpazesini destekler. Android için 4.4.2'ye kadar fiziksel imaj alımlarını destekler (sürüm 3 hariç). DS minimum sistem gereksiniminin düşük olması nedeniyle herhangi bir cihazda çalışabilir. Önemli kanıtlar için akıllı telefonun bellek dökümünü arayabilir [22].

4.2.9. MOBILedit! Forensic

MOBİLedit! Forensic [23], birkaç tıklamayla akıllı telefonda saklanan silinmiş veriler de dahil olmak üzere tüm verileri almak, aramak ve görüntülemeye izin verir. Bu araç, Android ve iOS tarafından desteklenen tüm akıllı telefonları destekleyebilir. Sıklıkla güncellenir ve daha fazla akıllı telefonu desteklemek için yeni özellikler eklenir. Araç, bu kanıtların elde ediliş şekli ve sunulması biçimini değiştirmiştir. *Mahkeme salonunda sunulmaya hazır ayrıntılı adli raporlar üretir*. Rapor herhangi bir dilde üretilebilir.

4.2.10. ViaExtract

ViaExtract [24] ViaForensics tarafından oluşturulan fiziksel ve mantıksal bir çıkarma aracıdır. Android akıllı telefonlar için rehberli veri toplama, güçlü analiz ve esnek raporlama özellikleri sunar. ViaExtract, yalnızca bir düğmeyi tıklatarak çoğu akıllı telefonun kök erişimini elde etmek için cihaz root'lama sihirbazını kullanır. Bu araç, incelemecilerin dahili ve harici depolamadan veri çıkarmak için şifreyi kırmalarına izin verir. Hızlı ve kullanımı kolay bir global arama özelliği sunar. Bu özellik, denetleyicinin, halihazırda açık olan tüm incelemelerde çıkarılan tüm içerik türlerini bir kerede aramasına izin verir. ViaExtract popüler Android akıllı telefonların çoğunda çalışır.

4.2.11. Examiner Plus (MPE +)

MPE + [25] gelişmiş akıllı telefon edinme ve analiz özelliklerine sahip bir mobil cihaz inceleme aracıdır. Geniş platform ve cihaz yelpazesini desteklemektedir. İncelemecilerin veriyi hızlı bir şekilde toplamasına, kolayca tespit etmesine ve etkili bir şekilde elde etmesine izin verir. DS gibi, MPE + da önemli bir kanıt için bir akıllı telefonun bellek dökümünü arayabilir [22]. Piyasada bulunan diğer araçlardan %30 daha hızlı iOS ve Android cihazlarından veri edinebilir. MPE +, sağlam ve üstün bir analiz araçları kümsesi içerir. Fiziksel edinimi gerçekleştirmek için, hedef telefona takılması gereken boş bir adli SD kartı, MPE + 'nın ajanını geçici olarak saklar. Root yetkisi kazanmak için 3. Parti araçlara ihtiyaç duyar.

5. FİRMWARE PROTOKOLÜNE DAYALI EDİNİM

2014 yılının üçüncü çeyreğinde Android işletim sistemi, akıllı işletim sistemi pazar payının yaklaşık %84'ünü oluşturdu (Smartphone OS Pazar Payı 2014 yılının 3. çeyreği). Android akıllı telefon pazarının boyutu artık PC pazarınınkini aşıyor, sürekli olarak kişisel ve iş kullanımı için çeşitli teknolojiler ortaya çıkıyor (Bring your own device, 2014). Bu eğilim, silinen dosyaların geri getirilmesine ve analiz edilmesine yardımcı olmak için flash belleğin fiziksel olarak edinilmesine yönelik araştırmaların özellikle gerekli olduğu Android forensics'in önemini de arttırmaktadır.

Mevcut Android fiziksel edinim yöntemleri şu sorunlara sahiptir: İlk olarak, çoğu adli araç, Android çekirdeğinin güvenlik açıklarını suistimal ederek veya özel image (işletim sistemi) (Rooting (Android OS), 2014; Vidas ve diğerleri, 2011) kullanarak akıllı telefonlardan veri toplamaktadır. Bununla birlikte, bu güvenlik açıkları sürekli olarak kapatılmakta ve çoğunlukla fiziksel bellek dökümü zafiyetlerinin giderildiği daha güvenlikli sürümler haline gelmekte. Ayrıca güvenlik teknolojilerinin son uygulamaları (Secure boot, 2014; Samsung KNOX, 2014) akıllı telefonlardan veri edinmeyi daha da zorlaştırıyor. İkinci olarak, özel kurtarma görüntüsünün (recovery image) değiştirilmesine dayanan döküm yöntemi (Son ve ark., 2013), kullanıcı verilerinin bütünlüğünü dikkate alan tek yaklaşımdır. Bununla birlikte, bu yöntem, özel kurtarma görüntüsünü cihaza yazmayı gerektirdiğinden, tüm flash bellek dökümünün bütünlüğünü garanti etmemektedir. Üçüncüsü, mevcut adli araçlar, akıllı telefonlardan veri edinmek için Android Hata Ayıklama Köprüsü (ADB) protokolünü kullanıyor. Bu nedenle, ekran deseni veya kullanıcı şifresi ile kilitlenmiş akıllı telefonlardan veri edinmek zordur (USB hata ayıklama devre dışı). Bu sorunları çözmek için, Android akıllı telefonların firmware güncelleme protokollerini analiz etmeye dayanan yeni bir fiziksel edinim yöntemi önermekteyiz.

Yazılım (S/W) tabanlı ve donanım (H/W) tabanlı edinme yöntemleri esas olarak Android akıllı telefonlardan veri edinmek için kullanılmaktadır. S/W tabanlı edinme yöntemleri, mantıksal edinim ve fiziksel edinim olarak ikiye ayrılmıştır. Mantıksal edinim yöntemleri, bir akıllı telefonda depolanan kullanıcı verilerini ADB Yedekleme (Android Backup

Extractor, 2014) veya İçerik Sağlayıcı (Hoog, 2011) aracılığıyla edinir. Bununla birlikte, bu yöntem yalnızca arama geçmişi ve resimler gibi kayıtlı dosyaları alır ve silinen dosyaları elde etmek için kullanılamaz. Fiziksel edinme yöntemleri, genel olarak USB kablosunu taktıktan sonra verileri doğrudan akıllı telefonun flash belleğinden çıkarır. Flaş belleğinin fiziksel olarak dökümünü gerçekleştirmek için öncelikle bir yönetici ayrıcalığı edinmek için root'lama işlemi gerçekleştirilmelidir. Root'lamaya dayalı edinim çalışmaları, bilimsel çalışmalarda ortaya konmuştur (Hoog, 2009; Lessard and Kessler, 2010). Bu çalışmalar, HTC akıllı telefonlarını geliştiren ve ADB kabuğu kullanan edinim yöntemleri de dahil olmak üzere Android adli bilişim konularını tartışır. Bununla birlikte bu yöntemler yalnızca USB hata ayıklama modu etkinleştirildiğinde veri edinmek için kullanılabilir. Ticari adli araçlar (Oksijen Forensic, 2014; AccessData MPE+, 2014; MSAB XRY, 2015) de bu yöntemi kullanmaktadır. Ancak, akıllı telefon açıldıktan sonra root'lama işlemi gerçekleştirildiğinden; Veri edinildiğinde bütünlük zarar görür. Buna ek olarak, Android işletim sistemi yeni bir sürümle güncellendiğinde, root'lamaya izin veren mevcut güvenlik açıklarına düzeltme eklenir ki bu nedenle, Android OS güncellendiğinde yeni bir root'lama tekniği bulunmalıdır. Cellebrite UFED 4PC (2015), kötüye kullanma yoluyla (exploit) temelde bir ADB fiziksel bellek dökümünü desteklerken bazı Samsung modelleri, özel bir önyükleme yükleyicisi aracılığıyla fiziksel bellek dökümünü desteklemektedir. Bununla birlikte, bu yöntemde, her modelin fiziksel bellek dökümü için ortak bir yükleyici yüklemek yerine farklı bir önyükleyici yüklenmesinin gerektiği bir sorunu vardır.

Bununla birlikte, USB hata ayıklama genellikle devre dışı olduğu için, desen kilidi veya kullanıcı şifresi ayarlandıysa bu yöntem geçerli değildir. Dahası, Secure Boot ve Samsung KNOX teknolojileri son zamanlarda Android'e uygulandığında, gelecekte bu edinim yöntemini kullanırken zorlaşacak olan özel imajların yazılması konusunda kısıtlamalar gelecektir.

Flash cihazları (RIFF Box, 2014, ORT aracı, 2014; Z3X box, 2014) mobil cihazlardan veri çıkarmak için de kullanılır. Bununla birlikte, bu araçların ana işlevi S/W hasarına uğramış kırılmış telefonları düzeltmektir. Dolayısıyla bu araçlar genel adli araçlar olarak düşünülmez.

H/W tabanlı edinme yöntemleri, JTAG tabanlı edinimi (Kim ve ark., 2008; Breeuwsma ve ark., 2007) ve Chip-off tabanlı edinimi (Jovanovic, 2012) içerir. JTAG tabanlı erişim yöntemi, akıllı telefonun PCB kartı üzerindeki JTAG hata ayıklama ara yüzünü kullanarak verileri flaş bellekten okur ve kopyasını çıkartır. Chip-off tabanlı yöntem, flash bellek yongalarını akıllı kartların PCB kartından fiziksel olarak ayırmayı ve flash belleğin ham verisini kopyalamayı temel alır. JTAG tabanlı edinme yöntemi sorunludur çünkü tüm akıllı telefonlar JTAG soketine sahip değildir ve veri edinmek uzun sürer. Chip-off tabanlı bilgi toplama yöntemi, flash belleği ayırdığı için sınırlı durumlarda kullanılır ve uygulanması ayrı bir uzmanlık gerektirir.

Flash bellek, esasen akıllı telefonlara veri depolamak için kullanılır. Flaş bellek fiziksel olarak küçük olduğundan ve çok miktarda veri depolayabildiğinden, akıllı telefonlar ve gömülü cihazlarda yaygın olarak kullanılmaktadır. Son zamanlarda, NAND flaşının ve bir denetleyiciyi ile aynı pakete entegre edildiği bir gömülü Multi-Media Card (eMMC), EXT4 dosya sistemini kullanarak depolanan verileri yönetmektedir. Ayrıca, BOOT, RECOVERY, SYSTEM ve USERDATA gibi bölümleri kurar ve çalıştırır. Tüm flash belleğin fiziksel dökümünden önce bir yönetici ayrıcalığı edinilmelidir. Yönetici ayrıcalığını elde etmek için BOOT bölümünde özel bir image üzerine yazılır ve bir uygulama (SuperUser.apk) veya bir binary dosya (*/system/su*) SİSTEM bölümüne kaydedilir. Buna ek olarak, yönetici ayrıcalığı, kurtarma modunda elde edilen root yetkisi ile veya Android işletim sistemindeki güvenlik açıklarından yararlanma yoluyla elde edilir. Genel olarak, Google'ın FASTBOOT (Android software development-fastboot, 2014) adlı özel bir imajı flash'a yazma için kullanılır. Her bir üretici kendi firmware güncelleme programlarını (Samsung Kies, 2014; Samsung Odin, 2014; LG Yazılım ve araçları İndirme, 2014; Pantech SelfUpgrade, 2014; HTC Sync Yöneticisi, 2014; Sony PC Companion, 2014; Xiaomi Xiaomi Smartphone için MiFlash'i İndirin) , 2015) kullanır. Google tarafından sağlanan FASTBOOT aracılığıyla basitçe firmware yazılmasını önlemek için bir protokol yayınlanmamış olduğundan sadece orijinal üretici yazılımı flash'a yazabilir. Ürün yazılımı güncelleme işlemi, yalnızca akıllı telefonlar yazılım güncelleme veya indirme modu adı verilen özel bir mod'a girdiğinde çalışır. Bu modda yalnızca ön yükleyici ve USB işlevi çalışabilir ve yeni bir sistem firmware'i (ya da işletim sistemi) yazılabilir. Güncelleme işlemleri ve komutları, IDA Pro (HexRays, 2015) gibi

bir araç kullanarak önyükleyici (boot loader) ve firmware güncelleme programının tersine mühendisliği ile analiz edilebilir.

5.1. Ürün Yazılımı Güncelleme Protokollerine Dayalı Android Fiziksel Edinimi

Adli bakış açısından, kullanıcı verilerini içeren flash bellek, veri edinimi sırasında ana hedeftir. Bu işlem, mantıksal bir edinim yöntemi yerine tüm flash belleği elde etmek için fiziksel bir edinim yöntemini gerektirir. Üretici yazılımı, güncelleme işlemi sırasında, Android OS ya da S/W sorunlarını gidermek için flash belleğine yazılır. Flaş belleğine doğrudan S / W aracılığıyla erişmenin tek yolu bir firmware güncelleme protokolüdür ve bundan dolayı firmware güncelleme işleminde kullanılan komutları analiz ederek yeni bir fiziksel bellek edinme yöntemi türetebiliriz.

Şimdiye kadar var olan adli edinim/imaj alma araçlarının sorunlarını çözmek için yazılım güncelleme protokollerini analiz eden bir araştırma yoktu. Bu çalışmada, LG, Pantech ve Samsung akıllı telefonları tarafından kullanılan üretici yazılımı güncelleme protokollerini analiz ettik. LG ve Pantech modellerinde, flash belleğe doğrudan erişim ve yazma komutlarının yanı sıra flash belleğin kopyasını çıkartmak için kullanılabilecek okuma komutlarının da yer aldığını gördük. Samsung modellerinde, flash belleğin dökümünü almak için daha önceden okuma komutlarının bulunduğunu doğruladık, ancak yeni versiyonlarda kaldırıldığını gördük. Bu analitik sonuçlara dayanarak, Android akıllı telefonlar için yeni bir fiziksel edinim yöntemi öneriyoruz.

5.2. Firmware Güncelleme Protokolü

Bir Android akıllı telefon açıldığında veya yeniden başlatıldığında, ROM'un 0. Adresinden itibaren yüklü bir proses çalıştırılır ve CPU yapılandırması da dahil olmak üzere başlatma işlemleri gerçekleştirilir. Sonra, önyükleyici belleğe yüklenir ve H/W (donanımlar) başlatılır ve NAND ve USB gibi bileşenler kullanım için yapılandırılır. Bootloader uygulaması, Initial BootLoader (IBL), Primary BootLoader (PBL), Secondary BootLoader (SBL) gibi bir çok aşamadan geçerek Android modeline bağlı olarak SBL önyükleme yükleyicisinde veya ABOOT önyükleme yükleyicisinde bir firmware güncelleme protokolü çalıştırılır.

Bir tersine mühendislik aracını kullanarak, önyükleme yükleyicisini analiz etmek ve firmware güncellemeleri için kullanılan komutları tanımlamak mümkündür. Yazılımı güncellemek veya flash belleğe erişerek veri edinmek için akıllı telefon normal önyükleme modundan ziyade yazılım güncelleme modunda olmalıdır. Bu modda yalnızca önyükleyici ve USB modülü etkinleştirildiğinden, edinilen verilerin bütünlüğü, fiziksel edinme işleminden sonra bile birçok kez garanti edilir. Dolayısıyla, incelemesi yapılan kanıt telefonu, güç kapalıyken bellenim güncelleme modunda önyüklenir ve daha sonra fiziki edinme yapılırsa, flaş belleğin bir görüntüsünü almak için bütünlük korunabilir. Şekil. 1, Samsung, LG ve Pantech akıllı telefonlar yazılım güncelleme modunda önyüklendiğindeki ekran görüntülerini göstermektedir. Üretici yazılımı güncelleme moduna girerken kullanılan yazılım güncelleme modunu ve yöntemlerini belirten terimler, üreticiler arasında farklılık gösterir.

Şekil 1 - Firmware güncelleme modları (Samsung, LG, Pantech)

Tablo 1, bir akıllı telefon önyükleme yapıldığında firmware güncelleme moduna girme yöntemlerini göstermektedir. USB Jig, akıllı telefonlar için firmware güncelleme moduna giren basit bir devredir. Samsung ve LG akıllı telefonlar, microUSB konektörünün 4 ve 5 numaralı pinleri arasında 300 K ve 910 KOhm'luk dirençlere (XDA geliştiricileri, 2012a, 2012b) göre yazılım güncelleme moduna girilebilir. Bu USB İzolasyon kabloları, Orijinal Donanım Üreticisi (OEM) kilidini açmak gibi mevcut sınırlamaların üstesinden gelmek için kullanılamaz.

5.3. LG Firmware Güncelleme Protokollerinin Analizi

LG akıllı telefonlar için, LG tarafından sağlanan önyükleyici ve güncelleme programını (LG Software & Tools Download, 2014) ayrıştırarak firmware güncelleme süreçlerini ve

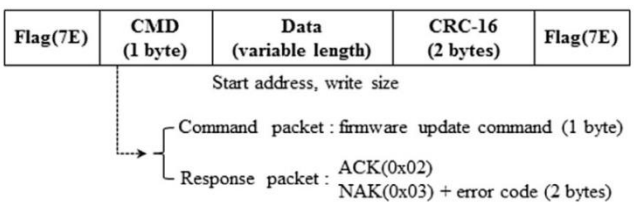
komutlarını analiz ettik ve flaşın kopyasını almak için kullanılan okuma komutunu tespit ettik.

5.4. LG Firmware Güncelleme Komutları

LG firmware güncelleme protokolü, bir komut paketi gönderme ve bir cevap paketi alma şeklinde çalışır. Paketler, HDLC bayrağı (0x7E) ve ardından paket veri ve Döngüsel Yedekleme Kontrolü (CRC) -16'dan başlayan ve HDLC bayrağı (0x7E) ile biten Üst Düzey Veri Bağlantısı Kontrolü (HDLC) çerçeve yapısını kullanır. Şekil. 2 LG telefonlarının firmware güncelleme komutunu yapısını göstermektedir.

Model	Modun adı	Tuş bileşimi
Samsung Galaxy	ODIN	Aynı anda, Ses Azaltma, Ev ve Güç tuşuna basıp basılı tutun (ardından Ses Seviyesini Artır tuşuna basın) veya 300 Kohm USB Jig bağlayın
LG Optimus	DOWNLOAD	Sesi Açma tuşunu basılı tutun telefonu mikroUSB kablosuyla bilgisayara takın veya 910 K ohm USB Jig bağlayın
Pantech Vega	PDL Download	Ses Seviyesini Arttır, Ses Azalt, Ev ve Güç tuşuna aynı anda basılı tutun
Google Nexus 4/5	DOWNLOAD	Sesi Açma tuşunu basılı tutun telefonu mikroUSB kablosuyla bilgisayara takın veya 910 K ohm USB Jig bağlayın.

Tablo 1 – Firmware güncelleme (download) moduna girme yöntemleri



Şekil 2 – LG Firmware güncelleme komut yapısı

LG firmware güncelleme moduna girdikten sonra, cihaz bilgilerini, GUID Partition Table (GPT) bölüm bilgilerini ve hafıza bilgilerini edinme komutları kullanılabilir. Tablo 2 LG firmware güncelleme komutlarını göstermektedir.

LG firmware güncelleme işlemleri aşağıdaki gibidir.

1. Cihaz bilgisini alın: 0x00.
2. İndirme moduna geç: 0x3A.
3. Sorgu özellikleri (Protokol ver., Vb.): 0x2F.
4. Partition bilgisini alın: 0x30.
5. Fabrika bilgilerini alın: 0xFA.
6. Sektör yazın (Birincil GPT): 0x39.
7. Bölümü yazmaya devam edin: 0x39.
8. İndirme tamamlandı: 0x38.
9. Sistemi yeniden başlatın: 0x0A.

5.4.1. LG flash bellek döküm komutu

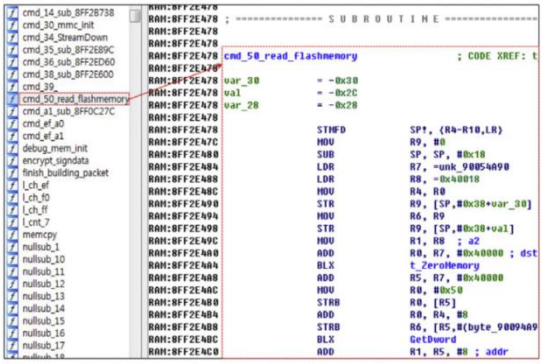
Önyükleme yükleyicisinde firmware güncelleme komutlarının tersine mühendisliği ile elde edilen sonuçlara dayanarak, Tablo 2'de gösterilenlere ek olarak flaş bellek için okuma komutlarını belirledik. Şekil 2 LG Optimus G modelinde (LG-E975) SBL3 önyükleme yükleyicisinin tersine mühendislikle elde edilen flash bellek için okuma komutunu (0x50) göstermektedir.

Firmware güncelleme moduna Tablo 1'de açıklanan işleme göre girildikten sonra, flash bellek ve GPT bölüm bilgisi boyutunu elde etmek için flash bellek bilgisi edinme (0x30) komutu gönderilir. Elde edilen bilgi, flash bellek boyutunu, bölüm sayısını, başlangıç adresini, bitiş adresini ve her bir bölümün adını içerir. Flash bellek (0x50) için okuma komutu başlangıç adresi ve döküm boyutu ile gönderilir ve daha sonra istenen boyut kadar flaş bellek verilerisi elde edilebilir.

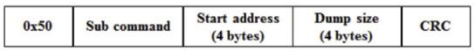
Komut	**Açıklama**
0x00	Cihaz bilgisini al (model, derleme tarihi)
0x3A	Download moduna git

0x2F	Protokol ve algoritma versiyonunu getir
0x30	MMC ve bölüm tablosu bilgisini getir
0xFA	Fabrika bilgilerini getir (IMEI, MAC adresi)
0x12	RAM belleği oku
0x39	Flash belleği yaz
0x0A	Sistemi resetle

Tablo 2 - LG firmware güncelleme komutları



Şekil 3 - . LG SBL3 ön yükleyicisinin tersine mühendislik yapılması



Şekil 4- LG okuma komutu formatı (0x50)

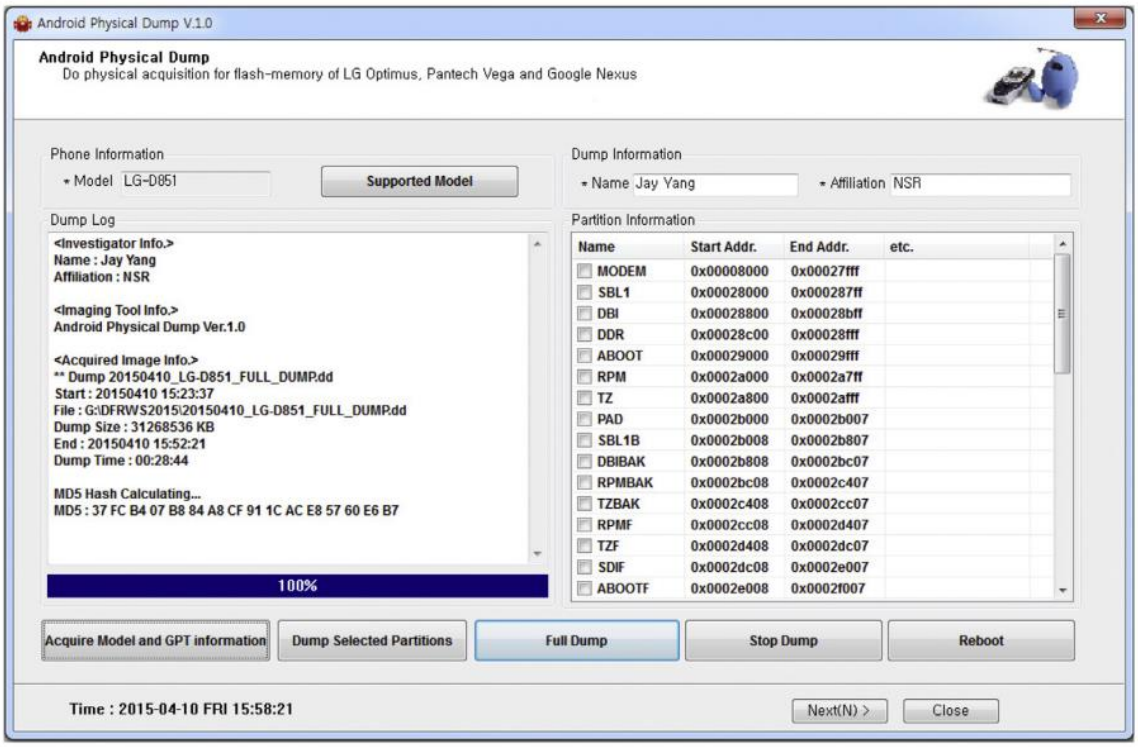
Şekil 4, flaş belleğin içini okuma komutunun (0x50) biçimini gösterir. Şekil 5, bir veri toplama örneğini göstermektedir. Tüm modellerin fiziksel olarak edinilmesi, Android işletim sisteminden ve çekirdek sürümünden bağımsız olarak okuma komutu kullanılarak gerçekleştirilebilir. Dahası, yazılım güncelleme modunda önyükleme yapıldığından, döküm görüntüsünün bütünlüğü daima korunur.

5.5. Android'in Fiziksel Edinimi

Makalede önerilen edinim yöntemini kullanarak, C++ ‘da Android Physical Dump (APD) adlı bir edinim aracı geliştirdik. Şekil 10 APD aracını göstermektedir. Bir USB kablosuyla bağlandıktan sonra fiziksel olarak edinme butonları tıklanır.

5.6. Desteklenen Modeller

APD aracı şu anda en yeni Android modellerinin 80'inden fazlasını destekliyor. Şu anda, APD, LG Optimus, Pantech Vega ve Google Nexus modellerinden fiziksel çöplükler yapabilir. Desteklenen modellerin temsili örnekleri şunlardır: LG G3, G2, G, Pantech R3, Iron2, Nexus 4/5 ve G Watch.



Şekil 10 – Android Physical Dump (APD)

5.7. Firmware Güncelleme Modunda Başlatma

İlk fiziksel döküm adımında, cihaz yazılım güncellemesi modunda önyüklenir. Akıllı telefonlar, AT komutu veya indirme modu komutu aracılığıyla firmware güncelleme moduna girebilir. Bununla birlikte, fiziksel bellek döküm alma işlemi, normal bir önyükleme sonrasında gerçekleştirilirse, edinilen verilerin hash değeri değişebilir. Bu

nedenle, telefon ele geçtikten sonra kapatılması ve ürün yazılımı güncelleme moduyla önyüklemesi yapılmalıdır.

5.8. Telefona Bağlanma ve Model Bilgisini Edinme

Bir USB kablosu kullanarak akıllı telefona bağlayın. USB sürücüsü önceden kurulmuş olmalıdır. USB sürücüsü üreticinin web sayfasından indirilebilir. Bir USB kablosu bağladıktan sonra, Model ve GPT bilgilerini al düğmesini seçin. Kullanıcı yalnızca üreticiyi seçerse, akıllı telefon model adı otomatik olarak görüntülenir.

Model bilgilerini aldıktan sonra, GPT bölüm bilgisi komutu gönderilerek bölüm bilgisi elde edilir. Şek. 10'da, pencere her bölüm için bölüm adını, başlangıç adresini ve bitiş adresini sunar.

5.9. Fiziksel Edinim

APD aracı, bir bölüm dökümünü ve tüm flash bellek dökümünü gerçekleştirmek için uygulanmıştır. Seçilen Bölümleri edinme düğmesi tıklandıktan sonra, ilgili bölümün fiziksel bir dökümü gerçekleştirilir. Döküm işlemi bittikten sonra, döküm görüntüsünün döküm süresi ve MD5 hash değeri Döküm Günlük penceresinde görüntülenir.

Full Dump butonu tüm flash belleği almak için seçilir. Fiziksel edinme işlemi, edinilen GPT bölüm bilgisindeki flash belleğin başlangıç ve bitiş adreslerini kullanarak yürütülür. Edinim işlemi tamamlandıktan sonra, araştırmacı bilgileri, edinim aracı bilgileri ve döküm bilgisi Şekil 10'da gösterildiği gibi görüntülenir. Döküm bilgileri, hesaplanan MD5 hash değerini ve döküm görüntüsünün başlangıç zamanı ve bitiş süresini gösterir. Hash değerinin hesaplanması, döküm imajının bütünlüğünü kontrol etmek için önemlidir.

APD aracını kullanarak elde edilen dosya ham veri formatıdır ve akıllı telefon adli araçları (Cellebrite UFED Fiziksel Analiz Cihazı 2015, Rehberlik EnCase, 2014, R-Linux, 2014) aracılığıyla analiz edilebilir.

5.10. Denemeler

Android adli bilişim alanında en önemli faktörler, desen kilidi ve kullanıcı şifresi nedeniyle adli imajının bütünlüğünü, hızlı kanıt toplama ve alınan flash imajının bütünlüğünü garanti altına almaktadır. Bu üç faktöre dayanarak, bu çalışmada önerilen APD aracını en yeni Android akıllı telefonlarını kullanarak mevcut edinim yöntemleriyle karşılaştırdık. Karşılaştırılan araçlar arasında, özel kurtarma görüntüsünü temel alan iyi bilinen Cellebrite UFED 4PC, döküm yöntemi ve root'la istismarı yoluyla ADB fiziksel dökümü ve JTAG tabanlı edinim vardı. Tablo 5, elde edilen deneysel sonuçları göstermektedir. G3 (F400S, D851), Optimus G (F180S, E975), R3 (IM-A850S), Iron2 (IM-A910S) ve Nexus 4/5 (E960, D821) gibi farklı modeller kullanılmıştır. Flaş belleğini APD aracı ile dışarı alma işlemini tamamladıktan sonra, alınan imajın kalitesini belirlemek için elde edilen imaj Cellebrite UFED Fiziksel Analiz Cihazı (2015) kullanarak karşılaştırılmıştır.

5.11. Elde Edilmiş Görüntünün Bütünlüğünü Koruma

Bir önceki çalışmada (Son ve ark., 2013), kullanıcı verileri bütünlüğü bir JTAG tabanlı edinim yöntemi ile kontrol edildi. Bu yöntem, kullanıcı verilerinin bütünlüğünü sağlar, çünkü yalnızca özel imajı flash belleğe yazılmıştır. Bununla birlikte, flash belleğin kurtarma bölümü değiştirildiğinden, tüm flash belleğin bütünlüğü hasar görür. Edinim işlemi sırasında kullanılan Cellebrite UFED 4PC ile bütünlük de hasar görüyor çünkü edinim işlemi normal açılıştan sonra gerçekleştirilmektedir.

Buna karşılık, önerilen erişim yöntemi, tüm flash belleğin bütünlüğünü korumaktadır. Yazılım güncelleme modunda önyüklenir ve fiziksel bilgi edinme, flash bellek okuma komutunu kullanarak gerçekleştirilir. Böylece, tüm flash belleğin bütünlüğünün muhafaza edilip edilmediğini teyit etmek için JTAG tabanlı edinim yöntemini karşılaştırrılabilir. Sonuçların doğruluğunu sağlamak için, deney işlemi bir önceki çalışmada olduğu gibi yerine getirilmiştir (Son ve ark., 2013). Beş kez flash bellek edinimi gerçekleştirildikten sonra edinilen görüntülerin karma değerleri karşılaştırılmıştır.

5.12. Edinme Hızı

Akıllı telefonların kullanımındaki hızlı artış nedeniyle, analiz edilmesi gereken akıllı telefonların sayısı da hızla artıyor. İncelenmesi gereken telefonlarının sayısı büyük olduğundan, alanda 8 saati aşan bir edinme zamanı gerektiren JTAG tabanlı erişim yöntemini kullanmak kolay değildir. Bu nedenle, hızlı S/W tabanlı edinim yöntemleri sıklıkla kullanılır. Tüm flash bellek için edinme süresini karşılaştıran sonuçlar Tablo 5'te gösterilmiştir. Sonuçlar, 8 modelin edinme sürelerinin ortalama değerlerini temsil etmektedir. Önerilen yöntem, verileri bir bilgisayara göndermek için akıllı telefonun maksimum boyutunu ayarlayabildiğinden, diğer yöntemlerle edinme süresinden yaklaşık dört kat daha hızlıdır. 32 GB flaş bellek elde etmek için ortalama olarak 30 dakika gerekiyordu.

	Açıklanan yöntem	**Cellebrite UFED 4P**	**Özel kurtarma modu**	**Root'larak ADB ile döküm alma**	**JTAG yöntemi**
Bütünlük garantisi	0	0	X	X	0
Imaj alma hızı (32GB)	30 dk.	120 dk.	120 dk.	180 dk.	480 dk. [a]
Ekran kilitli akıllı telefonun imajı alma	0	0	X	X	0

[a] JTAG tabanlı edinme yönteminde cihazı sökme ve adaptörü bağlantı süresi hariç tutulmuştur

Tablo 5 – Deney sonuçları

5.13. Ekranı Kilitli Akıllı Telefonlardan Fiziksel Olarak Edinme (USB Debug devre dışı)

Çoğu S/W tabanlı adli araç, fiziksel edinim için ADB protokolünü kullanır. ADB protokolünü kullanmak için akıllı telefonda USB hata ayıklamasının etkinleştirilmesi gerekir. Bununla birlikte, tüm Android akıllı telefonlar güvenlik nedenlerinden ötürü varsayılan olarak USB hata ayıklaması devre dışı bırakılmış olarak teslim edilir. Böylece, mevcut kazanım yöntemlerini uygulamak için USB hata ayıklamasının etkinleştirilmesi gerekir. Bu nedenle, bir desen veya kullanıcı şifresi tarafından kilitlenmiş (USB hata ayıklamalı devre dışı bırakılmış) bir akıllı telefondaki mevcut yöntemlerden birini kullanarak fiziksel edinimi gerçekleştirmek imkansızdır. Bu eksiklik, Android'in fiziksel edinimi alanında çözülmesi gereken önemli bir konudur. Bununla birlikte, önerilen yöntem bu sorunun üstesinden gelmektedir. Ekranı kilitli bir akıllı telefon olsa bile, telefon kapalıyken ve firmware güncelleme modunda yeniden başlatıldıktan sonra fiziksel olarak edinme gerçekleştirmek mümkündür.

Forensic Aracı	Açık Kaynak	Kullanıcı Dostu	Ekran Kilitli Cihazdan Kurtarma	Bütünlük	Bölüm	Dışarı Veri Aktarma	Adli Bilişim Süreçleri Uyumu	Genel Amaçlı
LiME	E	H	Bilgi yok	Yüksek	H	TCP SD Kart	H	H
AMExtractor	E	H	H	Yüksek	H	TCP	H	H
APD	H	H	E	E	E	Bilgi yok	H	H
Hawkeye	H	H	H	E	E	Bilgi yok	H	H
ANDROPHSy	E	H	E	E	E	TCP	E	E
ADA	E	E	H	E	H	SD Kart	H	H
UFED	H	H	E	E	H	SD Kart, USB Flash Bellek	E	H
Oxygen	H	E	E	E	H	USB bağlantısı, Bluetooth	E	H
MSAB XRY/XACT	H	E	E	E	H	USB Bağlantısı	E	H
DS	H	E	E	E	H	USB Bağlantısı	E	H
MOBILedit!	H	E	H	E	H	USB Kablo, TCP	E	H
ViaExtract	H	E	E	Bilgi Yok	H	Bilgi Yok	E	H
MPE+	H	E	E	H	H	Kablolar, Infrared, Bluetooth	E	H

Tablo -1: Adli Bilişim Araçlarının Karşılaştırmalı Tablosu

6. MOBİL CİHAZ YÖNETİM (MDM) YAZILIMI İLE DELİL ELDE ETME

MDM , Mobile Device Management (Mobil Cihaz Yönetimi) ifadesinin kısaltması olup, işletmelerin bilgi güvenliği için kullandığı en önemli teknolojidir. Mobil Cihaz Yönetim yazılımı akıllı telefonların yaygınlaşması ile birlikte kurumların cihazlarının güvenliğini sağlamak için tercih ettiği yazılımlardan olmuştur. Bir MDM yazılımı cihazın uzaktan izlenebilmesine ve yönetilebilmesine olanak tanır. Diğer taraftan MDM, akıllı telefon endüstrisinin bıraktığı güvenlik boşluklarına hitap eden hızla gelişen bir satış sektörüdür. Hükümetler ve endüstrideki mobil cihaz eğilimleri göz önüne alındığında, bir işletmede akıllı telefonu güvence altına alma gerekliliği ortaya çıkmaktadır. Bir kuruluşta güvenilmeyen cihazlara izin verilmesinin doğasında olan riskler nedeniyle MDM sistemlerine olan ihtiyacı artmaktadır. Kuruluşlar, mobil cihaz risklerini azaltmak için mobil cihaz güvenlik politikalarının sıkı bir şekilde uygulanmasına ihtiyaç duymaktadır.

MDM, dağıtım, güvenceye alma, izleme ve entegrasyon işlemlerinin yapıldığı idari alandır ve işletmelere temel olarak, iş yerlerindeki mobil aygıtların (akıllı telefon, tablet, vb.) yönetimine dair hizmetler sunar. Aşağıdaki işlevler sayesinde işletmeler, kullanıcıların çalışmalarını kesintiye uğratmadan, kablosuz bağlantı yoluyla mobil cihazları hızlı ve etkin bir şekilde yönetebilir:

a) Güvenlik Yönetimi: Mobil cihazın çalınması veya kaybedilmesi halinde, güvenlik önlemi olarak tüm kurumsal veriler uzaktan kaldırılabilir ve silinebilir.
b) Politika Yönetimi: Kurumsal verileri (kamerayı kapatmak, ekran görüntüsü aldırmamak, ekran kilidi ve parola kilidini zorunlu kılmak, vb. yoluyla) güvence altına almak amacıyla işletmelere yönelik bilgi güvenliği düzenlemelerini mobil cihazlara otomatik olarak dağıtır.
c) Yazılım Dağıtımı: İşletmelerin, uyguladıkları bilgi güvenliği politikasını esas alarak, kurulum için gerekli uygulamaları mobil cihazlara dağıtmasına olanak sağlar ve tekil kurulum yapma zorluğundan kurtarır.
d) Envanter Yönetimi: Cihazdaki yazılım ve donanım verilerini düzenli olarak toplar ve kullanıcıların uygulamaları nasıl kullandıklarını izler.

Birkaç lider MDM ürünü üzerine yapılan araştırma, kurumsal düzeyde Android cihazlardan otomatik olarak adli veri koleksiyonları elde etmede genel bir özellik eksikliği ortaya çıkardı. Bu verilerin bulunması, olayların yanıtı, güvenlik denetimi, proaktif güvenlik izleme ve adli soruşturmalar da dahil olmak üzere kuruluşlar arasında bulunan ortak güvenlik uygulamalarına yardımcı olacaktır. 2010/2011 Bilgisayar Güvenlik Enstitüsü Bilgisayar Suçları ve Güvenliği Anketi'ne göre, çeşitli şirketlerden gelenlerin %61.5'i iç denetimlerin bir güvenlik mekanizması olarak kuruluşlarında yapıldığını bildirdi. Buna ek olarak,%44 veri-kayıp önleme ve kullanıcı içerikli izleme programlarının bulunduğunu bildirmiştir (Richardson, 2010). Bu istatistikler, birçok organizasyonun içeriden öğrenebilecekleri tehditlerle ilgili kurumsal risklerin farkında olduğunu ve bunları hafifletmek için gerekli önlemleri aldıklarını göstermektedir; Ancak, Android akıllı telefonu izlemek için teknoloji eksikliği göz önüne alındığında, bu cihazlarda gerçekleştirilen birçok işlem denetlenmemektedir. İzleme seçeneklerinin olmaması, bu verilerin iç soruşturmalarda yaratacağı önemli etkiyle birlikte, bu çalışmanın konusu olan DroidWatch adlı bir prototip çözüm önerisi ve geliştirilmesine yol açtı.

Bu makale, politika ihlalleri, fikri mülkiyet hırsızlığı, yanlış kullanım, zimmete para geçirme, sabotaj ve casusluk da dahil olmak üzere iç soruşturmalar için kullanışlı verilerin toplanmasını otomatikleştiren bir Android uygulamasının ("uygulaması") tasarlanması ve uygulanması üzerine odaklanmaktadır. Veriler, Gingerbread 2.3.6 çalıştıran bir Samsung Galaxy S II Epic 4G Dokunmatik Android akıllı telefonundan toplandı. Ardından PHP, MySQL, Apache ve Splunk çalıştıran uzak bir Ubuntu sunucusuna gönderildi. Anti-virüs, kök algılama ve uygulamanın sonlandırılmasına veya kaldırılmasına karşı korunma gibi özellikler, sistem ve kurumsal güvenlik için gereklidir, ancak bu araştırmanın kapsamı dışındadır. Uygulanan çalışma ile toplanan tüm veriler, kurumsal veya resmi ağlarda yaygın olanlara benzer bir kullanıcı onayı vasıtasıyla gerçekleşir. Veriler, köklü ayrıcalıklar veya Android mimarisinin kullanılmasıyla elde edilemez.

6.1. Android Uygulama Geliştirme Terminolojisi

Android uygulama bileşenleri, bir uygulamanın davranışını tanımlamaya yardımcı olan Android çerçeve bloklarından oluşur (framework blocks) (Google. (N.d.). Uygulama

temelleri). DroidWatch içinde şu uygulama bileşenleri kullanılmaktadır: etkinlikler, hizmetler, içerik sağlayıcıları, yayın alıcıları, içerik gözlemcileri ve alarmlar. Aşağıda açıklanan her bileşen, farklı ve kullanışlı bir amaca hizmet eder.

6.1.1. Etkinlikler (Activities)

Bir kullanıcı arabirimini uygulayan bağımsız ekranlardır. Bilgi görüntüler, kullanıcı etkileşimini ister ve diğer etkinlikleri başlatırlar (Google. (N.d.). Uygulama temelleri). DroidWatch'da etkinlikler nadiren kullanılır; Genel kullanıcı deneyimini etkilememek için, çalışmaların çoğu arka planda gerçekleşir.

6.1.2. Hizmetler (Service)

Kullanıcı etkileşimi gerektirmeyen uzun süre devam eden işlemlerdir. Etkinlikler gibi diğer uygulama bileşenleri, diğer uygulamalar ve hizmetler çalışırken bile hizmet başlatabilir ve devam ettirebilir (Google. (N.d.) Uygulama temel bilgileri). DroidWatch, veri koleksiyonlarını ve aktarımları gerçekleştirmek için sürekli olarak bir servis kullanır.

6.1.3. İçerik sağlayıcıları (Content Providers)

Uygulama verisinin erişimini ve paylaşımını yöneten uygulama bileşenidir. Ön tanımlı tekil kaynak tanımlayıcıları (URI) aracılığıyla içerik sağlayıcılarıyla arabirimlenerek kullanılır (Google. (N.d.). Uygulama temelleri). DroidWatch içerik sağlayıcıları iki şekilde kullanır:

1. Diğer uygulamalar içinde saklanan verileri okurken
2. DroidWatch uygulaması içinde depolanan verileri okuyup yazar iken

6.1.4. Yayın alıcıları (Broadcast Receiver)

Bir Android cihazdaki yayın sistemi olaylarını işleyen ve bunlara yanıt veren uygulama bileşenleri (Google. (N.d.). Uygulama temelleri) 'dir. Gelen Kısa Mesaj Servisi (SMS) mesajları ve uygulama yüklemesi gibi olayları tespit etmek için DroidWatch da kullanılırlar.

6.1.5. İçerik gözlemcileri (Content Observers)

İçerik sağlayıcılarla ilişkili olduğunda, hedeflenen bir veritabanı altında yatan veri kümesinin içeriği değiştiğinde bildirim alırlar (Google. (N.d.) ContentObserver). DroidWatch bunu verilerin gerçek zamanlı değişikliklerini algılamak için kullanır.

6.1.6. Alarmlar (Alarms), periyodik olarak içerik sağlayıcıları sorgulamak ve yeni veriler çekmek için DroidWatch'ta yapılandırılan, cron işlerine (cronjobs) benzer şekilde zamanlanmış işlemlerdir. Güvenilirdir ve yalnızca belirlenen zamanlarda çalışırlar.

6.2. Android Uygulama Güvenliği

Google'ın Android uygulamaları güvenlik modeli, bir uygulamanın AndroidManifest.xml dosyasında (daha sonra "AndroidManifest" olarak anılacaktır) izin beyanını içerir. Varsayılan olarak, istenen izinlere sahip olmayan bir uygulama, "diğer uygulamaları, işletim sistemini veya kullanıcıyı olumsuz yönde etkileyecek her hangi bir işlemi gerçekleştiremez" (Google. (N.d.) İzinler). Bu, bir uygulamanın diğer uygulamaların özel verilerine erişememesi, şebeke servislerini kullanamaması, dahili / harici hafızaya yazamaması veya diğer temel işlevleri yerine getirememesi anlamına gelir. Yeni indirilen bir uygulama, yüklenmeden önce kabul edilmiş olması için kullanıcıya bildirilen izinlerini sunmalıdır. Bu durum Android 5 ile değişmiştir. Yeni güvenlik modeline göre önemli (tehlikeli kategorideki) izinlerin çoğu uygulama ilgili izini gerektiren bir işlem yaptığı sırada talep edilir. Böylece işletim sistemi çalışma zamanında kullanıcıdan ilgili işlem için izin vermesini ister.

6.3. Kökleme (Rooting)

Köklendirme, kullanıcıların normal kullanıcı kipi altında normalden daha yüksek ayrıcalıklı işlevler gerçekleştirmesine olanak tanır. Yasal veya gayri meşru amaçlar için kullanılabilir. Kullanıcılar, güvenlik kısıtlamalarını atlamak veya DroidWatch gibi bir uygulama aracılığıyla toplanan verilere müdahale etmek isteyebilir. Kök erişimi meşru olarak da kullanılabilir.(J. Grover / Digital Investigation 10 (2013) S12-S20 S13 adli araştırmacılar tarafından bir cihazdan veri çıkarılması) Ancak, mümkün olduğunca bundan kaçınılmalıdır. Süreç tipik olarak belirli bir aygıtta veya işletim sisteminde bir güvenlik açığını kullanır ve daha fazla güvenlik açıklarına neden olabilir. Köklendirme, bir cihazın bölümlerini de değiştirir (adli

bilişim uygulamalarıyla çelişen bir eylem); Bununla birlikte, gerekli koşulların ve verilerin türüne bağlı olarak köklenme kaçınılmaz olabilir (Vidas ve diğerleri, 2011). Kök erişimi, DroidWatch gibi bir uygulamanın özellik sayısını artırabilir; bunun sonucu olarak sistemin güvenliğini zayıflatabilir, birlikte çalışabilirliği düşürebilir ve akıllı telefon sağlayıcının garantisini tehlikeye sokabilir.

6.4. Akıllı Telefon Araştırmaları

Suçları araştırmaya ve hassas hükümet bilgilerini yetkisiz erişimden korumak için yetkilendirilmiş mobil güvenlikte birincil oyuncular kolluk kuvvetleri ve devlet kurumlarıdır. Şirketler ayrıca ticari casusluk, finansal hırsızlık ve fikri mülkiyet hırsızlığına karşı kendilerini korumak için mobil güvenlikle çok ilgilidirler. Boşanma kararları, velayet savaşları, emlak anlaşmazlıkları vb. alanlardaki özel menfaatler de bu alandaki ilerlemelerden kazançlı çıkmaktadır (Hoog, 2011). Sonuç olarak, akıllı telefonların izlenmesinden menfaat sağlayacak soruşturma türleri, kanun uygulama soruşturmaları, iç soruşturmalar ve özel soruşturmalardır. Bu araştırma, potansiyel politika ihlallerini, fikri mülkiyet hırsızlığını, kötüye kullanım, zimmete para geçirme, sabotaj, casusluk ve diğer soruşturmaları araştırmak için bir organizasyonda sözleşmeli veya başka bir şekilde (örneğin, adli olay inceleyicileri, güvenlik denetçileri vb.) personel tarafından gerçekleştirilen dahili soruşturmalar üzerine yoğunlaşmaktadır. Dahili araştırmacıların kolluk soruşturmalarının sıkı adli muamele ve koruma prosedürlerine uymaları gerekmez, ancak genellikle yaygın olarak uygulanan adli bilişim tekniklerine ve kurallarına uymaya çalışılmalıdır. Dahili araştırmalar için değerli akıllı telefon verileri elde etmek için, geleneksel olarak bir cihaza fiziksel olarak erişmek gereklidir. Buna bir istisna olan EnCase Enterprise, Ekim 2012 tarihinden itibaren bir ağ üzerinden Android cihazların uzaktan adli görüntülerini çıkarabilimektedir. Bazı MDM'ler ayrıca sınırlı izleme, ancak araştırmacıların ihtiyaçlarını etkin bir şekilde ele alacak kadar yeterli değildir. Bir mobil cihazın fiziksel olarak alındığı varsayılarak, araştırmacılar, cihazın mevcut durumunun mantıksal veya fiziksel anlık görüntüsünü almak için çeşitli araçlar kullanabilirler. Andrew Hoog, (Hoog, 2011), Android akıllı telefonlardan bilgi toplamak için mevcut araçların çoğunu listeliyor. Cihazların bazıları taşınabilir donanım aygıtları ve diğerleri yazılım ürünleridir; Bununla birlikte, hepsi evrensel bir seri veri yolu (USB) bağlantısı üzerinden çalışır ve çalışması için

akıllı telefona fiziksel erişim gerektirir. Buna ek olarak, araçların birçoğu kök erişim gerektirir (Valle, 2013).

6.5. Gizlilik endişeleri

DroidWatch, kullanıcıları gizlilik beklentileri hakkında bilgilendirmek ve onaylarını almak için bir telefonun önyükleme işlemi sırasında bir kullanıcı onayı bayrağı görüntüler (izin kartı afişinin uygulanmasına ilişkin daha fazla ayrıntı Bölüm 4.1.2'de bulunur). Bu, uygulamanın bir casus yazılım sınıflandırmasını önlemesine yardımcı olur ve sistem hatalarını engelleyebilir. Tablo 1'de, geçerli bazı mobil cihaz gizlilik davaları listelenmiştir. Kullanıcıları DroidWatch'ta izleme politikaları hakkında bilgilendirmek için istenen izinlerin kabul edilebilir kullanımı, ABD v. Ziegler'de (ABD Temyiz Mahkemesi, 2007) karar veren ABD Temyiz Mahkemesinden alınabilir. Potansiyel BYOD etkileri ile ilgili argümanlar, ABD Büyük Anayasa Mahkemesi davası Ontario v. Quon kararından (ABD Yüksek Mahkemesi, 2010) alınarak yapılabilir.

Sonuç olarak, bir organizasyon, DroidWatch gibi izleme uygulamalarını, şahsen sahip olunanlar da dahil olmak üzere tüm kurumsal akıllı telefonlara kurma hakkına sahip olduğunu düşünebilir; çünkü telefonlar özel olarak kontrol edilen ağında çalışırlar. Carrier IQ davası, bu yazının yazıldığı tarih itibariyle halen beklemede olmasına rağmen, kullanıcı verilerinin kullanıcı onayı olmadan akıllı telefonda izlenmesinden kaynaklanabilecek yasal sorunlara örnek teşkil etmektedir (Davis, 2012).

6.6. Ticari MDM Ürünleri

Üçüncü taraf MDM ürünleri mobil cihazlarla bir şirketin genel güvenliğini artırır; Bununla birlikte, çoğu MDM'de derinlemesine kullanıcı izleme özellikleri yoktur. **Zenprise**, **AirWatch** ve **MobileIron** gibi bazı önde gelen MDM seçenekleri, sınırlı izleme yetenekleri (ör., GPS izleme ve SMS izleme) sunmaktadır, ancak dahili araştırmalara yardımcı olan diğer mevcut veri setlerini toplamayı başaramamaktadır. Araştırılan MDM ürünlerinin Juniper Pulse Mobil Güvenlik Paketi (v.3.0R3) en çok kullanıcı izleme yetenekleri sundu ve bu araştırmanın bir parçası olarak değerlendirildi. Bulgular, ürünün DroidWatch'ta kapsanan veri kümelerinin yaklaşık %50'sini topladığını gösterdi; Bununla birlikte, verilerin

depolanması ve Juniper kontrollü sistemler tarafından barındırılması gerekir. Bu, bir kuruluşun denetim verilerini dahili olarak saklama şartını engelleyebilir.

Kişisel "casus" uygulamalar (örn., **Mobistealth**, **StealthGenie**, **FlexiSpy** ve **Mobile Spy**) gibi piyasada bulunan diğer ürünler, DroidWatch ile aynı veri kümelerinin çoğunu toplayabilir, ancak yükseltilmiş ayrıcalıklar için gereksinimler ve eksiklikler gibi sınırlayıcı faktörlere sahiptir. Ayrıca kurumsal depolama ve analiz yetenekleri açısından kullanıcı bilgisi olmadan kişisel bilgiler toplayan, genellikle casus yazılım olarak sınıflandırılırlar (Juniper Networks, 2012).

U.S. v Ziegler (2007)	Kullanıcıların politikadan haberdar olması durumunda bir kuruluş kendi ekipmanını izleme hakkına sahiptir
City of Ontario v. Quon (2010)	Denetlemeler, bir çalışan tarafından ödenen ek ücret ödemeleri bile şirket tarafından sağlanan bir cihaz üzerinde gerçekleştirilebilir
Carrier IQ	Mevcut değil. Bekliyor.

Tablo-1: Mobil cihaz mahremiyet davaları

Uzaktan kurumsal adli delil toplama araçları da kuruluşun güvenliğini artırmayı hedeflemektedir**. Google Rapig Response** (GRR), adli araştırmacılara ve olaya müdahale eden kişilere, adli olarak bir ağ üzerinden çok sayıda makinadan kanıt elde etmesini sağlar (Cohen ve diğerleri, 2011). GRR'ye benzer ticari çözümler EnCase Enterprise, **AccessData Enterprise**, **F-Response Enterprise Edition** ve **Mandiant Intelligent Response**'dır. EnCase Enterprise, Android'i desteklerken, diğerleri şu anda bunu desteklemez. Sözü edilen uzaktan adli araçlar, verileri sürekli olarak toplamayıp depolamadıkları için DroidWatch'tan farklıdır. Bunun yerine, operatöre talimat verildiğinde verilerin bir kerelik fotoğraflarını çekerler. DroidWatch kodu bu araçların yeteneklerini genişletmek için kullanılabilir.

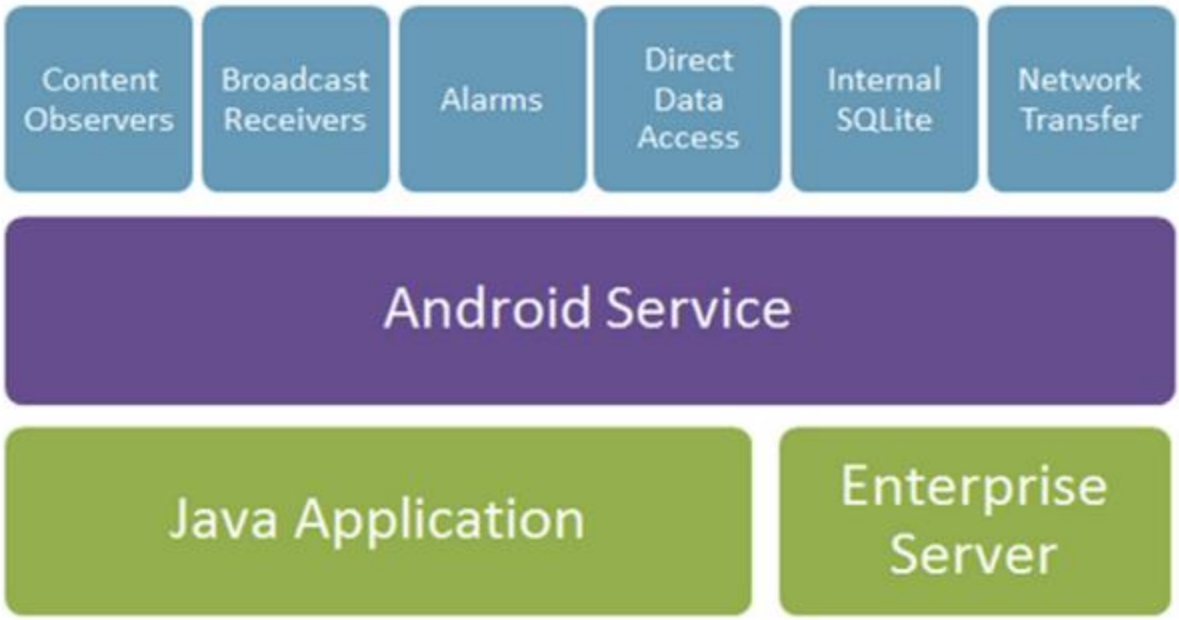
6.7. DroidWatch MDM Yazılımı

Bu araştırmayı çeşitli bilimsel çabalar şekillendirmiştir. Lee ve diğerleri tarafından önerilen bir sistem; bir akıllı telefondan Android Uygulama Programlama Arayüzünü (API) kullanarak aynı SDCard'a hızlıca veri çıkarmak için bir Secure Digital Card (SDCard) üzerinde bulunan bir Android uygulamasını kullanıyor (Lee ve ark., 2009). Verilerin

toplanmasına yönelik bu mantıklı yaklaşım, sisteminin bir cihazı sürekli olarak izlemesi dışında DroidWatch'a benzer. Bununla birlikte, çalışma sırasında kök ayrıcalıkları olmadan elde edilebilen çeşitli veri setlerini vurgulamaktadır. ViaForensics tarafından açık kaynaklı ürün olarak piyasaya sürülen AFLogical benzer bir yaklaşım benimser. Ayrıca, özel bir SDCard kullanır ve alınan veri kümelerinin sayısını genişletir (Hoog, 2010). Yang ve ark.'nın takip çalışmaları SDCards için bulut bilgi işleminin yerini almasını önerdi; Bununla birlikte, araştırmaları bir cihaza fiziksel olarak erişmeyi gerektirir ve sürekli veri toplamaz (Yang ve Lai, 2012).

Villan ve ark. , Sanal bir ağ bilgisine (VNC) benzer gerçek zamanlı izleme gerçekleştiren yerli bir Android uygulamasını, bir akıllı telefon üzerinde kök ayrıcalıkları kullanmadan gerçekleştirdi (Villan ve Esteve, 2011). Araştırma kullanıcıların ekranını kullanılabilirlik amacıyla uzak bir yere akıtmayı (yani, kurumsal yardım masalarında kullanmak için) akıtmayı içerir ancak bir kullanıcının ekranını izleyebilme özelliği DroidWatch'ta gelecekteki bir özellik olarak uygulanabilir. Shields ve ark. Tarafından sunulan araştırma. Yeni bir nesne parmak izi yaklaşımı (Shields ve ark., 2011) kullanarak sürekli ve proaktif bir şekilde bir ağ üzerinden gerçek adli bilişim edinimlerini gerçekleştiren ilk sistem olan Proaktif Nesne Parmak İzi ve Depolama (PROOFS) adında bir edinim ve izleme sistemi başlattı. PROOFS, Android akıllı telefonlarda çalışmazken, bir izleme aracının adli olarak kabul edilmesi gereken kriterlerini vurgulamaktadır. DroidWatch ile ilgili gelecekteki çalışmalar, bu kriterlerden bazılarının dahil edilmesini içerir.

Android platformundaki eski anti-adli bilişim çalışması, DroidWatch'ın nasıl tehlikeye atıldığını veya engelleneceğini değerlendirmede etkili oldu. Birkaç genel anti-adli kavramlar, Distefano ve diğerleri tarafından Android'e aktarıldı ve DroidWatch uygulamasının anti-adli değerlendirmesi sırasında bir rehber olarak görev yaptı (Bölüm 4.3) (Distefano ve ark., 2010). Azadegan ve arkadaşlarının yaptığı araştırma. Ek anti-adli bilişim konsepti sundu ve ayrıca yukarıda anılan DroidWatch değerlendirmesine dahil edildi (Azadegan ve ark., 2012).



Şekil-1: DroidWatch sistem mimarisi

Broadcast Receiver →Content Observer→Alarm

Şekil-2: Tasarım Stratejisi

Strateji, göreceli olarak kolay uygulanabilirlik, gerçek zamanlı bildirimleri işleme yeteneği ve yanlış pozitif ve çoğaltılması konularına odaklanmaktadır. İlk önce, sistem yayınları üretip üretmediğini belirlemek için veri setleri analiz edilmelidir. Eğer yaparlarsa, yayın alıcıları koleksiyonlar için uygulama bileşeni olarak düşünülmelidir. Yayınlar mevcut değilse, içerik gözlemcilerini uygulamaya geçirmeyi düşünün. Yayınlar ve içerik gözlemcileri hedeflenen veri koleksiyonları için kullanılamıyor veya etkisiz ise alarmlar kullanılmalıdır.

6.7.1. Yerel depolama

Tüm toplanan veriler telefonda yerel bir SQLite veritabanında geçici olarak saklanır ve sadece DroidWatch uygulaması tarafından erişilebilir olacak şekilde yapılandırılır. Standart Yapısal Sorgulama Dili (SQL) veritabanı fonksiyonları, özel bir DroidWatch içerik sağlayıcısı tarafından işlenir. Bu, her bir DroidWatch koleksiyonunun iş parçacığına göre güvenli ve yapılandırılmış bir biçimde gerçekleştirilmesini sağlar. Zamanlanmış bir alarm periyodik olarak yerel SQLite veritabanı dosyasını güvenli köprü metni aktarım protokolü (HTTPS) POST üzerinden işlenmek üzere kuruluş sunucusuna aktarır. Aktarım işlemi, veritabanı

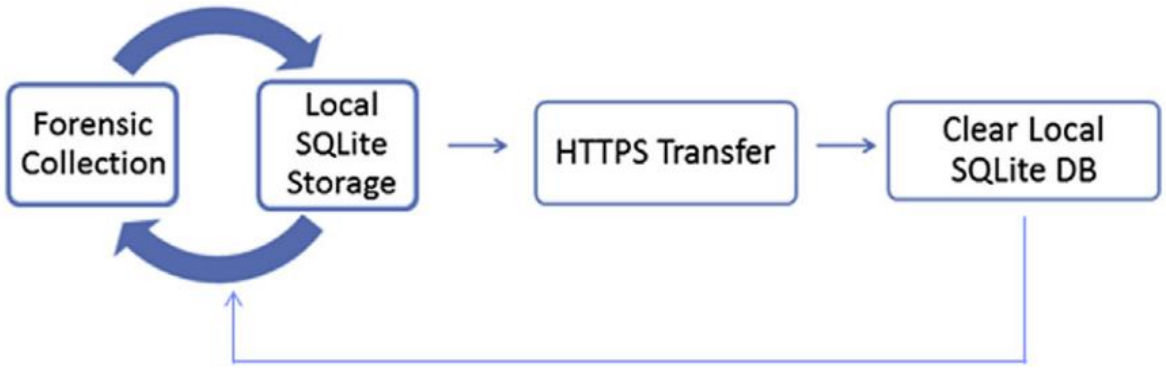
dosyasının nispeten küçük boyutunun (ortalama 75 kilobayt) yardımıyla bir kullanıcının deneyimine en az etkisi olan arka planda çalışacak şekilde tasarlanmıştır.

6.7.2. 6.7.8. Şirket sunucusu

Toplanan verilerin aktığı kurumsal sunucu prototipi, Apache, PHP, MySQL ve Splunk çalıştıran özel yerel bir ağdaki Ubuntu sanal makinesidir. Apache kendinden imzalı bir güvenli soket katmanı (SSL) sertifikasıyla yapılandırılmış ve DroidWatch uygulaması içinde bir varlık dosyası olarak dahil edilmiştir. Bu, bir HTTPS bağlantısı üzerinden veri aktarılmasına izin verir. PHP kodu, SQLite dosya yüklemelerini yönetir ve olayları bir MySQL veritabanına ayıklar. Splunk, periyodik olarak MySQL veritabanından veri çeker ve analiz ve raporlama için olayları arabiriminde kullanılabilir duruma getirir.

6.7.3. 6.7.9. Veri Akış Süreci

DroidWatch uygulaması içindeki veri akış süreci (Şekil 3) sürekli bir işlemdir, transferler her 2 saatte bir denenir (bu değer konfigüre edilebilir). Kurumsal sunucuya başarılı bir şekilde aktarıldıktan sonra, aktarmadan önce tarihli olaylar yerel telefon veritabanından silinir ve bu da o veritabanının boyutunu en aza indirir. Başarısız olan dosya aktarımları günlüğe kaydedilir ve herhangi bir etkinliğin silinmesine yol açmaz.



Şekil-3: Veri İşleme Akış Diyagramı

Cihaz hesabı bilgileri, DroidWatch hizmeti başlatıldıktan sonra doğrudan Android API üzerinden toplanır.

Data set	App component used		
	Broadcast receiver	**Content observe**	**Alarm**
App install/removal	x		
Browser navigation			x
Browser search			x
Calendar event			x
Call log		x	
Contact list		x	
Device accont[a]			
Device ID			x
GPS location			x
GPS location setting	x		
MMS	x		x
Pictre gallery		x	
Screen lock stats	x		
SMS	x	x	
Third-party app log			x

[a] *Cihaz hesabı bilgileri, DroidWatch hizmeti başlatıldığında doğrudan Android API aracılığıyla toplanır*

Tablo-2: Toplanan Veri Seti

6.7.4. Veri Kümeleri

Tablo 2, DroidWatch tarafından toplanan veri setlerini listeler. Bu veri setleri, mevcut içerik sağlayıcıları, iç araştırma için ihtiyaçlar ve erişilebilirlik seviyesine (yani kök gerekmez) bağlı olarak seçildi. Her veri kümesi, derleme aralıklarının ayarlanmasını sağlayan (yani, sistemin koleksiyonlar arasında ne kadar süre beklediğini) uygulama kaynak kodundaki bir varlık dosyası olan droidwatch.properties aracılığıyla yapılandırılabilir. Kuruluşlar, karşılık gelen aralık değerini sıfıra ayarlayarak bir veri kümesinin atlamasını seçebilirler. On beş benzersiz veri setine erişilebilir; İki veri kümesi ve hesap şifresi araştırılmıştır ancak kullanılmamıştır (aşağıda açıklanmıştır). E-posta uygulamasına erişmek için kullanılan mekanizmalar standart Android Yazılım Geliştirme Seti'nin (SDK) parçası değildir (CommonsWare, 2010). Buna ek olarak, e-posta uygulaması, üçüncü parti uygulamaların özel verilere erişmesini yasaklayan bir signatureOrSystem izniyle sınırlandırılmıştır (Android Open Source Project, 2008). Hesap şifreleri benzer korumalarla korunmaktadır; Arama uygulaması, AndroidManifest'te AUTHENTICATE_ACCOUNTS yetkisine izin vermeli ve kullanıcı kimliğini istediğiniz hesaba

(Google (N.D.). Hesap Yöneticisi) eşleştirmelidir. Bazı veri setleri, koleksiyonları gerçekleştirmek için birden fazla uygulama bileşeni kullanıyordu. Örneğin, Multimedya Mesaj Servisi (MMS) mesajları, bir yayın alıcısı ve bir alarm kullanılarak algılanır; Kullanılan bileşen mesaj odaklıdır.

6.7.5. Analiz ve Değerlendirme

Bu bölüm, DroidWatch denemesinin sonuçlarını açıklar ve bunları iç araştırmanın yardımcı olabileceği durumlara uygular. Senaryo dosyası deneyleri, mevcut veri kümelerine ve yazarın önceki iş deneyimine dayanır. Tüm sonuçlar tek bir cihazdan ve kullanıcıdan elde edilmiştir. Splunk'ın maliyeti (günlük 500 megabite kadar ücretsiz) nedeniyle bu araştırma için kullanıldığını unutmayın, diğer ürünler benzer işlevleri yerine getirebilir.
Şekil 4, Splunk'ta kaydedilen günlük olayların sayısını, veri seti ile ayrılmış olarak, tek bir gün aralığı boyunca göstermektedir. Örnek, bir cihaz tarafından üretilen 442 gün içeriyor ve kaydedilen nispeten az sayıda olayı vurguluyor. Günlük toplamlar kullanım alışkanlıklarına göre değişir; Ağır biçimde kullanılan bir cihaz, günlük toplamda bir artış görür. Deney sırasında kullanıcı deneyimi veya artan pil tüketimi üzerinde olumsuz bir etkisi yoktur.

6.7.6. Genel Kullanım Eğilimleri

Splunk'teki "Ekran Kilidi Açılmamış" arama, etkin telefon kullanımını gösteren kullanıcı eylemlerinin zaman çizelgesini (örn. PIN kodu, hareket veya parola girdileri) görüntüler. Birkaç güne kadar elde edilen sonuçlar Şekil 2'de görülebilir. Bu veriler belirli bir zamanda telefon etkinliğini belirlemek, maskeli kullanıcıları (ilk atanmış kullanıcı dışındaki kullanıcılar) bulmak veya çalışanlar için kullanım desenleri oluşturmak için kullanılabilir.

6.7.7. Şüpheli Kişiler ve İletişim

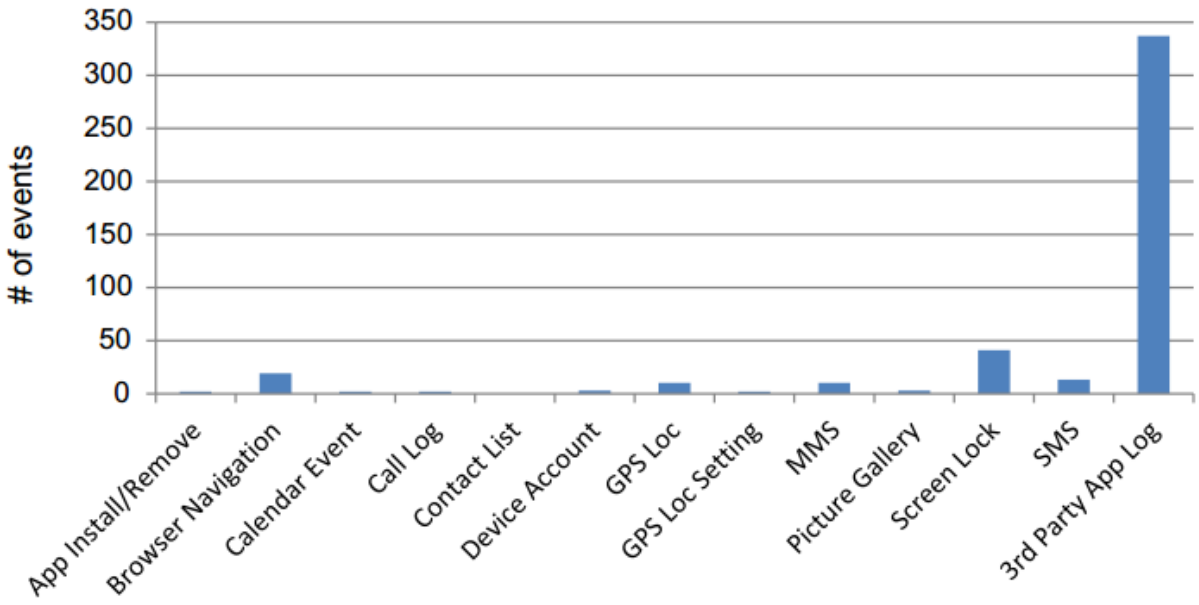
Bir kuruluşu riske atan kişilerin veya telefon numaralarının adlarını, arama yaparak veya bunları Splunk tetikleyicilerine (Splunk Enterprise'da bulunur) bulabilirsiniz. Bunlar, bir şirketin telefon defterinin dışındaki numaralar veya kara listeye girmiş insanlar olabilir. Kaydedilen SMS ve MMS içeriği de şüpheli etkinlikler için aranabilir. DroidWatch'te gelen

SMS, zaman damgasını ham SMS olarak işleyerek üçüncü parti bir zaman kaynağı olarak da hizmet edebilir. SMS yapısında gömülü olan zaman, telefonun zamanına bağlı değildir (Casey, 2009). Bu verilerin analizi sırasında karşılaşılan bazı sorunlar şunlardı:

Şekil-5 :Algılanan ekran kilidi açma aksiyonları (Splunk)

- Birden fazla kişiye gönderilen mesajlar, günlüklerde yalnızca bir alıcı listeledi.
- Gelen SMS'lerde saat dilimleri eksik.
- MMS mesaj metni mevcut değildi.

Bir fotoğraf cihaz kamerasıyla çekilip hemen MMS ile gönderildiğinde, etkinlik koşullara bağlı olarak daha fazla soruşturma yapılmasını gerektirebilir. Şekil 6, 22 Aralık 2012 Cumartesi günü 03.20'de çekilen bir resim günlüğünü gösteren bir aramayı göstermektedir. Kullanıcının ofiste ve yalnız olduğunu tespit edilirse (muhtemelen GPS izleme yoluyla) bir veri sızıntısının olup olmadığını belirlemek için daha fazla analiz gerekli olabilir.



Şekil-4: 24 saatte kaydedilen olaylar

Şekil-6: Fotoğraf ve MMS arama sonuçları

6.7.8. Konum izleme

DroidWatch tarafından kaydedilen bilinen son konumlar arasında cihaz kimliği, enlem, boylam ve yakalama süresi bulunur. DroidWatch'ın konumları toplamak için kullandığı yaklaşım pil ömrünü korur ancak kaydedilen yerlerin seyrek kaydedilmesine neden olur. Yedi günlük süre boyunca yalnızca dört yer olduğu bildirildi. GPS sağlayıcı ayarı etkinleştirilmiş olsa da, GPS aktif olarak kullanılmadıkça bilinen en son konumlar bir cihazda saklanmaz (diğer bir deyişle, Google Haritalar uygulaması mevcut konumu görüntülemek için açılır). Ayrıca bir telefonun son bilinen konum değeri, cihazın yeniden başlatılması üzerine silinir ve kaydedilen bir koordinat kümesinin kaydedilmeden önce kaybolmasına neden olur.

Konum sağlayıcı ayarında yapılan değişiklikler izleme için de kullanılabilir. Telefonun fiziksel konum verileri daha güvenilir hale gelirse, bu veriler potansiyel olarak yararlı olur. GPS ayarı manuel olarak kapatıldığında bir cihazın konumunun tanımlanmasına izin verir.

Kaydedilen takvim olayları araştırmacılara da faydalıdır. Bir kullanıcının randevuları için yapılan aramalar, geçmişteki kontrol, gözetim planlaması veya seyahat planlarını belirleme konusunda yardımcı olabilir.

6.7.9. İnternet geçmişi

DroidWatch, yerleşik Android Web tarayıcısı içinde gerçekleştirilen etkinlikleri toplar ve kullanılabilir duruma getirir. Bir İnternet geçmişi etkinliği, alınan eylemi (ör. Göz atma veya arama), arama terimini veya URL'yi, etkinlik saatini ve ilişkilendirilmiş cihaz kimliği

içerir. Bu bilgiler, bir şirkette şüpheli tarayıcı kullanımını tanımlamak için kullanılabilir (örneğin, fikri mülkiyetlerin harici web sitelerine yüklenmesi). Tarayıcı aramaları, tespit edilen işlemlerin ardındaki kullanıcının olası niyetlerini daha iyi tahmin edebilmek için ayrıştırılabilir.

6.7.10. Kötü amaçlı uygulamalar

Yüklü uygulamalar için bir denetim, bir cihazın kötü amaçlı yazılım veya diğer endişe uyarıları içerdiğini ortaya çıkarabilir. Bu, dahili bir soruşturma sırasında ek endişeler ve güvenlik önlemleri alınmasını garanti eder. Sağlanan DroidWatch sonuç alanlarına, uygulamanın adı, gerçekleştirilen eylem ve kurulum / kaldırma tarihi dahil. Üçüncü taraf uygulama günlükleri de DroidWatch tarafından toplanır. Birkaç filtreleme mekanizması, günlükleri, yalnızca telefonda yerleşik olmayan uygulamalar tarafından üretilenlere sınırlar. Filtreleme mükemmel olmadığı halde, toplanan uygulama günlüklerinin toplam miktarı, gün başına 337 günlük (10.000'i aşan) daha yönetilebilir bir ortalamaya düşürüldü.

6.7.11. Adli bilişim

İzleyen bölümler, anti-adli bilişim kategorileri, Android antiforensi alanında yapılan önceki çalışmalardan alınmış ve anti-adli olabilecek güvenlik açıkları için DroidWatch uygulamasını değerlendirmek için kullanılıyor. Kanıtları gizleme (Bölüm 4.3.2), kanıt kaynaklarını değiştirme (Bölüm 4.3.3), kanıtları taklit etme (Bölüm 4.3.4) ve adli bilişim araçlarını tespit etme (Kısım 4.3.5) kategorileri kanıtları yok ediyor (Bölüm 4.3.1) (Distefano ve diğerleri, 2010; Azadegan ve diğerleri, 2012). DroidWatch'in mevcut haliyle, kök saldırılarına, uygulamanın kaldırılmasına ve işlem sonlanmalarına karşı tamamen duyarlı olduğunu unutmayın. Kök algılama ve uygulama yükleme politikalarının uygulanması gibi MDM'ler tarafından sunulan dış koruma, DroidWatch'ta veri bütünlüğünü sağlamaya dayanır.

6.7.12. Delilleri yok etme

Yayın alıcıları ve içerik gözlemcileri aracılığıyla toplanan veri setleri, olay ortaya çıktıkça her olayın kopyası DroidWatch'ın özel saklama alanına kaydedildiğinden, kanıt imha yöntemlerine karşı muhtemel değildir. Bununla birlikte, kanıtları kaldırmak mümkün olabileceğinden (örneğin, giden MMS mesajları, üçüncü taraf uygulama günlükleri, takvim etkinlikleri, tarayıcı gezintileri, tarayıcı aramaları ve bilinen en son GPS konumları) alarmlardan alınan veri setleri tahribat taktiklerinden etkilenir. Bir sonraki planlanan koleksiyona başlamadan önce. DroidWatch ile ilgili endişe, uygulamaların yükleme sonrasında özel niyet filtresi öncelikleri için kayıt yapabilmesidir. Maksimum maksat filtre öncelik değeri olan 231-1 kullanan bir uygulama, başka bir uygulama tarafından yayınlanmadan önce yayını engelleme ve bırakma özelliğine sahiptir. DroidWatch, bir araştırma prototibi olarak durumundan dolayı varsayılan intent-filter öncelik değerini kullanır.

6.7.13. Kanıt gizleme

Zamanlanmış alarmlar yoluyla toplanan veri setleri veri gizleme taktiklerinden etkilenenler arasındadır. Son gönderilen birkaç MMS iletisini gizlemek isteyen bir kullanıcı, bunları DroidWatch toplama işleminden yönlendirmek için el ile aktarma yöntemlerini kullanabilir. Yukarıda bahsedilen uygulamalar için özel amaçlı filtre öncelikleri kaydetme özelliği, benzer bir şekilde, uygulamanın yayınların engellenmesi ve yeniden yönlendirilmesi yoluyla verileri gizleyebilmesini sağlar. Örneğin, meşru bir üçüncü parti SMS uygulaması olan GoSMS, gelen SMS mesajlarını aktarmak ve sistem bildirimlerinin çoğaltılmasını ortadan kaldırmak için olası en üst düzey filtre önceliğini kaydetmektedir (Kovacevic, 2011).

6.7.14. Kanıt kaynaklarını değiştirme

Kanıt kaynaklarını değiştirmek, bir veri kümesini bir toplama işlemini engellemek için değiştirmeyi içerir (Distefano ve diğerleri, 2010). Bu, DroidWatch için başka bir endişe alanı. Alarmlar tarafından toplanan veriler duyarlıdır çünkü süreçler, mevcut bir veri setinde belirli değerlere dayanır. Örneğin, yeni giden mesajlar için MMS içerik sağlayıcısı

aracılığıyla tarama yapılırken "msg_box" alanı, gönderilen / giden MMS'i temsil eden "2" ile iletinin yönünü belirtir. Bu alanın değeri "5" olarak değiştirilirse, ileti toplama işlemi sırasında yoksayılır.

6.7.15. Taklit kanıtları

Mobil cihazlardaki sahteciliğe dayalı kanıt, araştırmacıların kafasını karıştırmak veya kaçmak için mevcut veri setlerine hayali veri ekleme işlemlerini içerir. DroidWatch koleksiyonları bu açıdan savunmasızdır, çünkü sahte girişleri gerçek olanlardan ayırmak için herhangi bir kontrol gerçekleştirilmez. DroidWatch için bir başka endişe olan hizmet reddi saldırısında kısa sürede büyük miktarda hayali veri eklenmesi. Yeterli veri bir telefona yüklenirse, uygulamanın düzgün çalışması durabilir.

6.7.22. Adli bilişim araçlarının tespit edilmesi

Adli bilişim araçları tarafından gerçekleştirilen adli bilişim araçlarının araştırılmasının doğrudan DroidWatch için geçerli olmadığı tespit edildi (Azadegan et al., 2012). Onlar, Android telefonlarında bazı tanınmış adli bilişim araçlarının ilk bağlantı imzalarını dinlemeye odaklandı. DroidWatch, geleneksel adli bilişim araçlarının aksine izlemeyi gerçekleştirir ve bir cihaza fiziksel olarak erişmek için herhangi bir ilk bağlantı imzası veya gereksinimi yoktur. Bununla birlikte, imza tespit fikri, DroidWatch'ın tarifeli transferlerine uygulanabilir.

6.8. AndroidWatch İleri Araştırma

DroidWatch ile ilgili gelecekteki araştırmalar, uygulama ve kurumsal sunucu üzerinde çalışmayı içerir. Yaklaşan bölümler, ek veri setleri toplamak (Bölüm 5.1) ve anti-sabotaj mekanizmalarını uygulamak için önerilen gelişmeleri kapsar (Bölüm 5.2).

DroidWatch'ın gelecekteki iyileştirmelerinin yanı sıra DroidWatch'ın bir MDM çözümü içine entegrasyonu, Android güvenlik topluluğu için çok değerli olacaktır. Mevcut MDM sistemleri, dahili soruşturmalara yardımcı olabilecek kullanıcı izleme özelliklerine sahip değildir. Sağlam politikanın uygulanmasını, uzaktan cihaz yönetimini ve Android cihazlarda kapsamlı bir kullanıcı izlemesi kombinasyonu sağlayan genel kurumsal güvenlik sistemi,

Android akıllı telefon dağıtımlarını düşünen hükümet ve endüstri kuruluşları arasındaki güvenlik endişelerini azaltmaya yardımcı olacaktır.

6.9. Ek veri setleri

Android adli bilişimi, her yeni işletim sistemi sürümüyle değişen gelişen bir alandır. Yeni veri setleri ve özellikleri ortaya çıktıkça, muhtemel bir izleme sistemine dahil edilmeleri için bunların bir araştırmaya katma değeri değerlendirilmelidir. Gelecekteki DroidWatch içerikleri şunları içerir: USB hata ayıklama ayarları, telefon yeniden başlatma, sesli posta günlükleri ve dumpsys, dumpstate ve dmesg'den ek uygulama ve çekirdek günlükleri.

6.10. Koruma önleyici mekanizmalar

Toplanan ve bir telefonda saklanan veriler, kullanıcıların ve uygulamaların müdahale etmesini önlemek için halihazırda Android yerleşik güvenlik modeli (Kısım 2.2) üzerinde çalışıyor. DroidWatch'ı daha sıkı hale getirmek için bazı yetenek önermeleri şunları içerir:

- Veritabanı olaylarının şifrelenmesi (sağlama toplamı ile)
- Yüksek niyetli filtre öncelik değerleri
- Log günlüğünü tutma
- Etkinliğe dayalı koleksiyonlar ve transferler
- Veritabanı karması

DroidWatch, veritabanındaki olayları şifrelemek, kullanıcıların önceden toplanan olayları görüntülemesine veya müdahalesini engellemeye yarayan bir mekanizma. Her olay bir sağlama toplamıyla eşleştirilebilir ve bir ortak anahtar altyapısı kullanılarak şifrelenebilir (kurumsal sunucuda depolanan özel anahtar ile). AndroidManifest'e maksimum niyet filtre öncelik değerlerini kaydetmek, iki uygulamanın aynı öncelik değerine kaydolması durumunda ne olacağını belirlemek için daha fazla araştırmaya ihtiyaç duyulmasına rağmen, uygulamaların sistem yayınlarını engellemesine engel olabilir. DroidWatch veritabanına "canlı tutma" mesajlarının periyodik olarak günlüğe kaydedilmesi servis kesintilerini vurgulamaktadır. Kütükler arasında zamandaki boşluklar varsa kurcalamaya

neden olabilir. Olay tabanlı tetikleyiciler, daha rasgele bir aktarım kalıbı sağlayabilir ve zamanlanmış operasyonlara karşı zaman esaslı engeller girişimleri önleyebilir; Bununla birlikte, bu kabiliyetin etkinliği hakkında daha fazla araştırmaya ihtiyaç vardır.

7. SONUÇ

Fiziksel adli bilişim araçlarındaki artış ve Android akıllı telefonlarının pratik kullanımıyla birlikte, bu makale, yeni araçları mevcut araçlar ile karşılaştırmak isteyen araştırmacılar için ölçütler belirlemiştir. Aynı zamanda, araştırmacıları veya uygulayıcıları, güvenilir ve uygun fiziksel adli araçlar seçerek, fiziksel imajları elde etmek için daha etkili, interaktif ve uygun bir yol sağlar. Oxygen ve Cellebrite UFED gibi ticari adli araçlar, güvenilir, kullanımı kolay, birçok Android sürümünde kullanılabilir ve tüm adli bilişim süreçlerini destekler. Ancak, her Android akıllı telefonda, özellikle de hedef akıllı telefon kilitli olduğunda bunlar geçerli değildir. Dahası, kişisel kullanım için uygun değildir. Açık kaynak araçları ise genellikle kullanıcı dostu değil, sadece imaj edinim aşamasına odaklanır, ve Android akıllı telefonların sınırlı sürümünde kullanılabilir, ancak yine de güvenilirdir. Bir ilgi çekici açık kaynak kodlu adli bilişim aracı da ANDROPHSY, özellikleri ile ticari adli araçlar ile rekabet edebilir. Herhangi bir sürüm Android akıllı telefonun tüm belleğini elde etmek için kullanılmak üzere tasarlanmıştır. Bu mobil cihaz adli yaşam döngüsünü destekleyen ilk açık kaynak araçtır.

Android akıllı telefonların firmware güncelleme protokollerini analiz ederek tüm flash belleği elde etmek mümkündür. Üreticiler tarafından sağlanan firmware güncelleme protokolleri önyükleme yükleyicisindeki açıklıklar kullanılarak bu yapılabilir. Üretici yazılımı güncelleme protokollerinin analitik sonuçlarına dayanarak, bazı flash bellek güncelleme protokollerinde flaş bellek okuma komutlarının yer aldığını ve dolayısıyla fiziksel edinimin gerçekleştirilebilmesi mümkündür. Bu yeni edinme yöntemine dayanarak, 80'in üzerinde en yeni Android modellinin flash belleğinin fiziksel olarak dışarı aktarılmasını destekleyen bir edinme programı geliştirilmiştir. Bu araç mevcut yöntemlerle karşılaştırıldığında, yeni yöntemin tüm flash belleğin bütünlüğünü koruduğu ve yüksek hızla edinimi gerçekleştirildiği kanıtlanmıştır. En güzel yanı; edinip yapılırken bir desen veya kullanıcı parolası ile ekran kilitlemesine bağlı kısıtlamaya takılmaksızın fiziksel edinme yapılabilir.

Kötü yanı her yeni Android akıllı telefonun güncelleme protokolünün analiz edilmesi gerektiğidir. Bununla birlikte, üretici yazılımı güncelleme protokolü tüm Android akıllı telefonların önyükleme yükleyicisinde uygulanır ve diğer taraftan her üretici tüm modellerine aynı üretici yazılımı güncelleme protokolünü uygular. Bu şekilde edinme yöntemi analiz edilerek üreticilerin tüm modelleri için fiziksel edinme gerçekleştirilmesine izin veren yazılım güncelleme protokolü bulunabilir. Bu nedenle, bu alanda sürekli araştırma yapılması gerekmektedir.

Mobil cihaz yönetimi aracılığıyla, Android sistemin kök ayrıcalıkları olmaksızın kurumsal ortamlarda sürekli Android cihazların izlenmesi ve korunması mümkündür. DroidWatch, türünün ilk açık kaynak sistemidir; Ancak, yeteneklerini genişletmek ve geliştirmek için daha fazla geliştirilmesi gerekmektedir. Güvenliği artırmak için anti-sabotaj mekanizmalarının da uygulanması gereklidir. Belirtildiği gibi, toplanan veri setleri çeşitli nedenlerle çeşitli iç tetkik türleri için yararlıdır. Bu araştırma, Android uygulamaları bileşenlerini izleme için önceliklendirmek adına kullanılabilecek, yeni bir geliştirme tasarım stratejisine katkıda bulunmaktadır. Son olarak, bu çalışma, varsayılan Android API aracılığıyla erişilebilen veri kümelerine erişmek için bir rehber işlevi görür.

Son çalışmada ise Android mobil uygulamalarının gizliliğini incelenmiş ve değerlendirilmiştir. Özellikle, açık kaynaklı adli bilişim araçlarını kullanarak Android mobil cihazların uçucu belleğindeki kimlik doğrulama bilgilerinin keşfedilip keşfedilemeyeceğini gösterilmiştir. Sonuçların analizi, incelenen Android uygulamalarının çoğunun uçucu bellekteki kimlik doğrulama bilgilerini kurtarma konusunda savunmasız olduğunu ortaya koydu. Mobil bankacılık uygulamaları gibi güvenlik öncelikli uygulamaların bile savunmasız olduğu kanıtlandı. Dahası, uçucu belleğin yalnızca kimlik doğrulama bilgilerini içermediğini cihazı yeniden başlatıldığında veya pilini çıkartıldığında dahi gözlemlendi. Ayrıca, uygulamanın kimlik doğrulama bilgilerinin tam olarak bir bellek dökümünde nerede bulunduğunu gösteren kalıp ve ifadelerin varlığını kanıtlanmıştır. Son olarak, kullanıcıların çeşitli web sitelerinde ve uygulamalarda aynı şifreyi tekrar kullanma eğiliminde olduklarını göz önüne alarak; tüm geliştiricilerin, uygulamanın kritikliğine bakılmaksızın, doğru ve güvenli programlama teknikleri ve yönergelerini kullanmaları gerektiği sonucuna

varılmıştır. Kimlik doğrulama bilgisi keşfini engellemek ve mobil platformlar tarafından sağlanan gizlilik düzeyini arttırmak için, incelenen senaryolardan yararlanılmalıdır.

Referanslar

[1] T. B. Tajuddin and A. A. Manaf, "Forensic investigation and analysis on digital evidence discovery through physical acquisition on smartphone," in 2015 World Congress on Internet Security (WorldCIS), Dublin, 2015

[2] L. Cai, J. Sha, and W. Qian, "Study on forensic analysis of physical memory," in Proc. of 2nd International Symposium on Computer, Communication, Control and Automation (3CA 2013), 2013.

[3] H. Yang, , J. Zhuge, H. Liu, and W. Liu,"A tool for volatile memory acquisition from Android devices," in Advances in Digital Forensics XII, New Delhi, Springer International Publishing, 2016, pp. 365-378.

[4] J. Sylve, A. Case, L. Marziale, and G.G. Richard, "Acquisition and analysis of volatile memory from android devices," Digital Investigation, vol. 8, no. 3, pp. 175-184, 2012.

[5] I. Kollár, "Forensic RAM dump image analyser," MCS thesis, Charles Univ., Prague, Czech Republic, 2010.

[6] M. Guido, J. Buttner, and J. Grover, "Rapid differential forensic imaging of mobile devices," Digital Investigation, vol. 18, pp. S46-S54, 2016.

[7] L. Spector, "USB 3.0 speed: real and imagined," PCWorld,
2014. [Online]. Available: http://www.pcworld.com/article/2360306/usb-3-0-speedreal-and-imagined.html. Accessed: Oct. 17, 2016.

[8] C. A. Jayasinghe, "Android smart phone contact analyzer",
MSIS dessirtation, 2015. Android Open Source Project.

[10]Azadegan S, Yu W, Sistani M, Acharya S. Novel anti-forensics approaches for smart phones. In Hawaii International Conference on System Sciences (pp. 5424–5431). Maui, HI: IEEE; 2012, January 4.

[11] Casey E. Top 7 ways investigators catch criminals using mobile device forensics. http://computer-forensics.sans.org/blog/2009/07/01/top-7-ways-investigators-catch-criminals-using-mobile-device-forensics; 2009, July 1.

[12] Citrix. IT organizations embrace bring-your-own devices.
http://www.citrix.com/site/resources/dynamic/additional/Citrix_BYO_Index_report.pdf; 2011, July 22.

[13] Cohen MI, Bilby D, Caronni G. Distributed forensics and incident response
in the enterprise. In: Digital forensics research workshop 2011. New Orleans, LA: Elsevier;

2011S101–10; August 2011.

[14] CommonsWare. Access Android emails through content provider. http://stackoverflow.com/questions/3811608/access-androidemails-through-content-provider; 2010, September 28.

Gazi Gelecektir