|  |
| --- |
| **Olay Müdahale Süreçlerinde Linux Sistemlerden Manuel Veri Toplama** Mustafa Emre DEMİR1\*, Şengül DOĞAN2, Türker TUNCER3 1 Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Merkez, Elazığ, Türkiye  2 Fırat Üniversitesi, Teknoloji Fakültesi, Adli Bilişim Mühendisliği, Merkez, Elazığ, Türkiye  3 Fırat Üniversitesi, Teknoloji Fakültesi, Adli Bilişim Mühendisliği, Merkez, Elazığ, Türkiye  \*1191144101@firat.edu.tr, 2 sdogan@firat.edu.tr, 3 turkertuncer@firat.edu.tr |
| **(Geliş/Received: 30/04/2022; Kabul/Accepted: 08/08/2022)** |

**Öz:** Günümüzde gelişmiş kalıcı tehdit gruplarının (APT) yaptığı saldırılar gerek kurumları gerekse özel şirketleri hedef almaktadır. Bu tür APT gruplarının yaptığı saldırılar birçok farklı amaca hizmet edebilmektedir. Özellikle kamu kurumlarına karşı yapılan saldırılarda devletlere ait gizli bilgilerin elde edilmesi gibi hedefleri olduğu bilinmektedir. Bu tür atakların tespit edilmesi için kurumlarda var olan tüm uç noktaların merkezi bir yerden sürekli olarak takip edilmesi ve kurum/kuruluş içerisindeki tüm aktivitelerin kontrol edilmesi ihtiyacını ortaya çıkarmaktadır. Böyle bir sistem kurum veya kuruluşlarda kurulsa dahi sistemler üzerinde gerekli inceleme ve analiz işlemlerinin yapılması için insan kaynağı ihtiyacı ortaya çıkmaktadır. Ayrıca bu sistemler genellikle Windows odaklı çalışmaktadır ve Linux destekleyenlerin verimi oldukça düşüktür. Linux sistemler bu yapıda kör kalmaktadır. İzleme yapılsa dahi bahsedilen türde bir saldırıya maruz kaldığının tespit edilmesi ve incelenmesi mevcut kullanılan kurumsal çözümlerle başarılı olmamaktadır. Bu sebeplerle Linux sistemlerde olay müdahale süreçlerinin işletilmesinde bir zorluk olduğu değerlendirilmiştir. Bu makalede tespit edilen zorluğa çözüm sunulmaktadır. Olay Müdahale süreçlerinde Linux işletim sistemine sahip cihazlarda veri elde edilmesine yönelik çalışmalar yapılmış ve çıktıları açıklanmıştır. Özetle bu makalede sunulan çözüm yöntemi kullanılarak Linux tabanlı çoğu işletim sisteminde hızlı ve doğru sonuçlarla olay müdahale süreçleri tamamlanmıştır.

**Anahtar kelimeler:** Linux Adli Bilişim, Linux İçin Olay Müdahale, Manuel Linux Adli Bilişim Teknikleri, Olay Müdahale, Veri Toplama.

**Manuel Data Retrieval from Linux Systems in Incident Response Processes**

**Abstract:** Nowadays, Advanced Persistent Threat (APT) groups target different types of companies. Their purposes are to obtain confidential data of organizations. In order to detect such attacks, all endpoints in organizations should be monitored and analyzed from Security Operation Center Team. At the same time, human resources are required for the management of these systems. In addition, these systems generally work on Windows and solutions that support linux are less successful. For these reasons, it is considered that there is a difficulty in the operation of incident response processes in Linux systems. This article proposes a solution to the identified difficulty. In Incident Response processes, studies were carried out to obtain data on devices with Linux operating system and their outputs were announced. In summary, incident response processes were completed with fast and accurate results in most Linux-based operating systems using the solution method presented in this article.

**Key words:** Linux Forensics, Incident Response for Linux Forensic, Digital Forensics for Linux, Retrieval data from Linux operating system.

**1. Giriş**

Advanced Persistent Threat (APT), Türkçeye Gelişmiş Kalıcı Tehdit veya Gelişmiş Sürekli Tehdit olarak çevrilen terim, dünya genelinde profesyonel siber saldırı yapan gruplara verilen genel isimdir [1]. Gruplara özel bireysel isimleri mevcuttur. Örneğin Lazarus veya APT38 olarak isimlendirilen grubun Kuzey Kore ile ilişkili olduğu bilinmektedir [2]. Bu grup genellikle IT ve Kimya sektöründe çalışan firmaları hedef almaktadır. Günümüzde bu tür saldırı grupları yaygınlaşmıştır dolayısı ile bu tür saldırıların analizi için kullanılan yeni tekniklerin geliştirilme ihtiyaçları ortaya çıkmaktadır [3,4].

Adli bilişim ve olay müdahalesi süreçlerinde inceleme yapılan kurum alt yapısındaki tüm uç noktaların hızlı bir şekilde analiz edilmesi ve tüm tehdit unsurlarının hızlıca tespit edilip ortadan kaldırılması hayati önem taşımaktadır [5]. Bu incelemelerin temel amacı kurum içerisinde kalıcılık sağlamış olabilecek tehdit gruplarının tespit edilmesi ve olası veri kaçakçılığı gibi olumsuz sonuçlar doğurabilecek durumların ortadan kaldırılmasıdır [6]. Bu tür saldırılarla ve saldırganlarla mücadele etmek amacı ile birçok sistem geliştirilmiştir [7,8]. Geliştirilen sistemler çoğunlukla Windows işletim sistemlerine hitap etmektedir. Linux destekleyenler dâhi istenilen performansı vermemektedir. Olay müdahale süreçlerinde Linux sistemlerden veri elde etme süreçlerinde birçok problemle karşılaşılmaktadır. Bunlardan ilki, Linux sistemlerin birçoğuna güncel EDR (Endpoint Detection Response; bilişim sistemlerinde tehdit unsurlarının tespit edilip gerektiğinde aksiyon alınmasına imkân sağlayan teknolojik çözümlere verilen genel isimdir.) araçlarına ait ajanların desteklenmemesi ve Linux tarafındaki görünürlüğün sağlanamamasıdır. Bir diğer problem ise ajan destekleyen ürünlerde dahi tam fonksiyonlu olarak sistemlere erişim sağlanamaması ve istenilen verinin istenildiği anda elde edilememesidir. Bu sebeplerle Linux sistemlerde ilkel analiz yöntemlerine başvurulmaktadır. Linux sistemlerin analizi için birçok adli bilişim aracı kullanılmaktadır. Bu araçların kullanımı için gerekli lisans maliyetlerinin karşılanması, sistemin imajının alınması ve ayrı bir ortamda bahsedilen araçlar kullanılarak analiz edilmesi gereklidir. Bu araçların yaptığın incelemeler ilkel kalmakta ve olay müdahale esnasında süreçleri oldukça uzatmaktadır [9-11].

Yukarıda açıklanan problemlerin olay müdahale süreçlerinde analiz ve kök sebebin ortaya çıkarılmasını yavaşlattığı aşikârdır. Bu sebeple tüm Linux sistemlerden incelenmesi gereken sistem verilerinin otomatik şekilde elde edilmesini sağlayacak araç tüm süreçleri hızlandıracaktır. Bu kapsamda değerlendirildiğinde herhangi bir programlama dilinde yazılacak bir aracın gerekli dosyaları istenilen bir hedefe ayrıştırılmasını sağlayabilecek bir çözüm ile hali hazırda olan bir siber güvenlik olayına müdahale etmek oldukça hızlanacaktır.

Bu çalışmanın katkıları aşağıdaki gibidir.

1. Adli bilişim yazılımları hem yüksek maliyete sahiptirler hem de bazı delilleri paket adli bilişim yazılımlarıyla elde etmek mümkün değildir çünkü her adli bilişim yazılımının kendine has özelliği vardır. Bu çalışmada önerilen yöntemle amaca yönelik adli bilişim analizi yapabilecek scriptler yazılmıştır.
2. Adli analiz yazılımları genellikle yavaş çalışırlar (karmaşıklıkları yüksek). Bu çalışmada sunulan metotla düşük karmaşıklıkta Linux sistemler için adli analiz yapılabilecektir.
3. Bu çalışmada operasyonel olarak kullanılabilecek bir analiz aracı detaylı bir şekilde açıklanmıştır.

Bu çalışmada mevcut Linux komutları ve python ve bash programlama dilleri kullanılarak Linux sistemlerden hızlı ve pratik şekilde veri toplama ve inceleme yapılması hedeflenmektedir. Olay müdahale süreçlerinde analizlerin nasıl yapıldığının detaylı anlaşılması adına literatürde bu işlemlerin nasıl gerçekleştirildiği “2. Adli bilişim ve Olay Müdahalesi” başlığında açıklanmıştır. Normal şatlarda Linux sistemlerde adli bilişimin nasıl yapıldığı “3. Linux Sistemlerde Adli Bilişim” başlığında açıklanmıştır. Önerilen çözümde kullanılan yöntemler hakkındaki bilgiler 4. Bölümde verilmiştir. Bulgular Bölüm 5’te, değerlendirme ve sonuç ise Bölüm 6’da açıklanmıştır.

**2. Adli Bilişim ve Olay Müdahalesi**

Adli bilişim, dijital ortamlardan elde edilen delillerin, bir konuyu ortaya çıkarmak amacıyla kabul görülebilir şekilde farklı ortamlara aktarılması ve daha sonrasında elde edilen delillerin değerlendirme, inceleme ve raporlama süreçlerini kapsayan ve Adli Bilimlerin altında yer alan bir bilim dalıdır [12,13].

Olay müdahalesi ise bir kurum veya kuruluşun gelişmiş kalıcı tehditlere (APT) maruz kalması sonucunda, tüm sistemlerde var olan tehdit unsurlarının tespit edilmesi, olay ile ilgili kök nedenin ortaya çıkarılması, kalıcılık sağlayan tehdit unsurlarının tekrar oluşmamasını sağlayacak şekilde temizlenmesini ve sistemlerin yeniden güvenli hale getirilmesini sağlayan süreçler bütünüdür [14].

Olay müdahalesi süreçlerinde incelemeler yapılırken Adli Bilişim tekniklerine başvurulur. Olay müdahalesinde temel amaç tehdit unsurlarının en hızlı şekilde ortadan kaldırılıp sistemlerin yeniden güvenliğini sağlamaktır. Bu sebeple adli bilişim tekniklerinde kullanılan delil elde etme süreçleri olay müdahalesinde kullanılmamaktadır. Delil elde etme süreçlerinde başvurulan yöntemler işlemlerin yavaş yapılmasına ve tehdit unsurlarının kalıcılığının artmasına sebep olmaktadır. Dolayısıyla yapılması gereken en öncelikli işlem tüm sistem envanterinin çıkartılması ve sistemlerin hepsinin tekelden görünür hale getirilmesini sağlamaktır. Görünürlük sağlandıktan sonra sistemler üzerinde bulunan kritik logların kural setlerinden geçirilerek aktif yetkisiz işlemlerin tespiti amaçlanmaktadır [14].

Olay müdahalesi süreçlerinin işletilebilmesi için kurum ve kuruluşlarda bulunan tüm sistemlerin, işletim sistemi ve platform bağımsız olarak gerekli log kaynaklarında istenilen logların merkezi yönetim birimine aktarılması gereklidir. Olay Müdahalesi süreçleri ulusla arası kabul görmüş iki temel başlıkta ele alınabilir. Bu başlıklar “2.1 Adli Bilişim ve Olay Müdahalesi Süreçleri” başlığı altında detaylandırılmıştır.

**2.1 Adli Bilişim ve Olay Müdahalesi Süreçleri**

Adli bilişim, bilginin bütünlüğünü korurken ve veriler için katı bir gözetim zinciri sürdürürken verilerin tanımlanması, toplanması, incelenmesi ve analizi gibi işlem basamaklarını takip eder [8]. Kuruluşlar, birçok kaynaktan gelen sürekli artan miktarda veriyi kullanır. Adli bilişim teknikleri, Suçları ve dâhili politika ihlallerini araştırmak, bilgisayar güvenliği olaylarını yeniden oluşturmak, operasyonel sorunları gidermek, kazayla oluşan sistem hasarından kurtarma gibi birçok amaç için kullanılabilir. Sistemlerinde ve ağlarında hangi olayların meydana geldiğini belirlemek için pratik olarak her kuruluşun adli bilişim yeteneklerine ihtiyacı vardır. Olay müdahalesi planı, bir siber güvenlik olayına müdahale etmek için hazırlanmış talimatlardır. Bir güvenlik olayı ciddi sorunlar açabilecek ise, hasarı hızla kontrol altına almak, en aza indirmek ve sonuçlardan bir ders çıkartarak tekrar oluşmamasını sağlamak için belirli adımlar atılır [15].

Her siber güvenlik olayı, soruşturma gerektirmeyebilir. Şirket içindeki bir çalışanın tek bir giriş hatası gibi olaylar, münferit olaylar olarak meydana geldiğinde farkında olmak önemlidir. Siber güvenlik ekipleri, her türün ne zaman araştırılması gerektiğine ilişkin belirlenmiş sınırları olan olay türlerinin bir listesine sahip olmalıdır. Böylelikle, her olay türü için özelleştirilmiş olay müdahale adımlarına sahip olmanız gerekir [5]. Bu çalışmada olay müdahale süreçlerine değinilecek ve Linux sistemlerde yapılması gereken adımlar ele alınacaktır.

**2.1.1 SANS** **Olay Müdahalesi Adımları**

SANS Olay müdahale planı, bilişim uzmanlarının ve personelinin veri ihlali veya siber saldırı gibi bir siber güvenlik olayını tanımasına ve bunlarla başa çıkmasına yardımcı olan, altı farklı aşamadan oluşan belgelenmiş, yazılı bir plandır [16,17]. Bu Planlar sırası ile Tablo 1’de gösterilmiştir.

**Tablo 1** SANS Olay Müdahale Adımları

|  |
| --- |
| **SANS Olay Müdahalesi Adımları** |
| Preparation – Hazırlık |
| Identification – Tanımlama |
| Containment – Muhafaza Etme |
| Eradication – Temizleme/Yok Etme |
| Recovery – Yeniden Kullanılabilir Hale Getirme |
| Lessons Learned – Çıkarılan Dersler |

**2.1.2 NIST Olay Müdahalesi Adımları**

NIST, Ulusal Standartlar ve Teknoloji Enstitüsü anlamına gelir. Olay müdahale adımları ile olay müdahalesi için sektör standartlarından birisi haline gelmişlerdir. NIST’e ait olay müdahalesi adımları Tablo 2’de gösterilmiştir [18].

**Tablo 2** NIST Olay Müdahale Adımları

|  |
| --- |
| **NIST Olay Müdahalesi Adımları** |
| Preparation – Hazırlık |
| Detection and Analysis – Tespit ve Analiz |
| Containment, Eradication ve Recovery – Muhafaza Etme, Temizleme/Yok Etme ve Yeniden kullanılabilir Hale Getirme |
| Post Incident Activity – Olay Sonrası Aktiviteler |

**2.1.3 SANS’a Göre Olay Müdahale Adımlarının Açıklanması**

Sans’a göre olay müdahale adımları aşağıda verilmiştir [16,17].

**Hazırlık:** Hazırlık aşamasının amacı, kuruluşun bir an önce bir olaya kapsamlı bir şekilde müdahale edebilmesini sağlamaktır. Bunun için önceden hazırlanması gereken kritik unsurlar; Politika, Müdahale, Planı Stratejisi, İletişim, Belgelendirme, Ekip, Erişim kontrolü, Eğitim, Araçlar olarak söylenebilir.

**Tanımlama:** Bu adım, kuruluştaki normalde süre gelen işlemlerin tespit edilmesini ve anomali olarak değerlendirilmesi gereken aktivitelerin tespit edilmesi aşamasıdır. Bir aktivitenin şüpheli bir olayı temsil edip etmediğini anlamayı ve olayın ne kadar önemli olduğunu belirlemeyi içerir. Bu aşamada tüm BT altyapısındaki uç noktaların tespit edilip gerekli araç kurulumlarının yapılması süreçleri işlenir. Aynı zamanda günlük dosyaları, hata mesajları ve güvenlik araçlarından gelen uyarılar dâhil olmak üzere birden fazla kaynaktan gelen olayları analiz etme süreçleri işletilir. Birden çok kaynaktan gelen verileri ilişkilendirerek bir olayı tanımlama ve mümkün olan en kısa sürede raporlama süreçleri de bu aşamada işletilir.

**Muhafaza:** Muhafaza etmenin amacı, mevcut güvenlik olayından kaynaklanan hasarı sınırlamak ve daha fazla hasarı önlemektir. Olayı tamamen hafifletmek ve aynı zamanda kovuşturma için gerekli olabilecek delillerin yok edilmesini önlemek için işletilmesi gereken süreçleri içerir.

**Temizleme:** Temizleme, saldırıların ortaya çıkardığı kötü amaçlı yazılımları veya diğer yapıları tamamen ortadan kaldırmayı ve etkilenen tüm sistemleri tamamen geri yüklemeyi amaçlamaktadır. Temizleme aşamaları Tablo 3’te açıklanmıştır.

**Tablo 3** Temizleme Aşaması Süreçleri

|  |  |
| --- | --- |
| **Temizleme Aşaması Süreçleri** | |
| **Yeniden görüntüleme** | Herhangi bir kötü amaçlı içeriğin kaldırılmasını sağlamak için etkilenen sistem sabit sürücülerinin tamamen silinmesi ve yeniden imajının oluşturulmasıdır. |
| **Kök Sebebi Önleme** | Olayın nedenini anlamak, örneğin saldırgan tarafından kötüye kullanılan bir güvenlik açığına yama uygulamak gibi gelecekteki tehlikeleri önlemeyi içerir. |
| **Temel güvenlik uygulamalarını uygulama** | Örneğin, eski yazılım sürümlerini yükseltme ve kullanılmayan hizmetleri devre dışı bırakmayı amaçlamaktadır. |

**Yeniden Kullanılabilir Hale Getirme:** Kurtarma işleminin amacı, temiz olduklarını ve tehdidin ortadan kaldırıldığını doğruladıktan sonra tüm sistemleri tam çalışır duruma getirmektir. İşlemleri geri yüklemek için saat ve tarihin tanımlanması için sistem sahipleri, güvenlik ekibinden gelen bilgilere dayanarak hizmetlerin ne zaman geri yükleneceği konusunda nihai kararı vermelidir. Test etme ve doğrulama, sistemlerin temiz ve çalışır durumda olduklarını doğrulama adımıdır. İzleme, olaydan sonra işlemleri gözlemlemek ve anormal davranışları kontrol etmek için bir süre izlemeyi sürdürme adımıdır. Aynı olayın tekrarlanmasını önlemek için geri yüklenen sistemlerde neler yapılabileceğine karar verme aşamasıdır.

**Çıkarılan Dersler:** Olayın bitiminden en geç iki hafta sonra, Güvenlik Ekibi olay hakkındaki tüm ilgili bilgileri derlemeli ve gelecekteki olay müdahale faaliyetlerine yardımcı olabilecek dersleri çıkarmalıdır. Bir olayın devam ederken tüm yönlerini belgelemek gereklidir ve kapsamlı belgelere ulaşmak, bir dahaki sefere alınacak dersleri belirlemek için çok önemlidir. Bir olay raporu yayınlanmalı, rapor tüm olayın gözden geçirilmesini sağlamalı ve Kim, Ne, Nerede, Neden ve Nasıl sorularına cevap vermelidir. Güvenlik Performansını iyileştirmenin yollarının belirlenmesinde ise doğru şekilde işlenmeyen ve bir dahaki sefere iyileştirilebilecek öğelerin tespit edilmesi gerekir.

**3. Linux Sistemlerde Adli Bilişim**

Adli bilişim araştırmacıları, güvenilir kanıtları ortaya çıkarmak için birçok farklı sistem ve araca aşina olmalıdır. Bu nedenle, ister bir Linux makinesinde bir analiz yapmak ister Linux'u bir araştırma platformu olarak kullanmak olsun, Linux ile aşinalık gereklidir. Analiz yürütmek için bir platform olarak Linux, bir araştırmacının gerçeği ortaya çıkarmasına izin verecek çeşitli araçlar sağlar. Ayrıca, bir araştırmacının bir Linux makinesini denetlemek için gerekli olan yöntem ve kavramlardan haberdar olması da önemlidir. Ayrıca, Linux dosya sistemi günlüğünün yeterli bilgisi, bir Linux makinesinde kanıt bulmak için kritik bir faktördür [10].

Temel ve basit bir Linux dosya sistemi, önemli bir dosya bilgisini koruyabilir ve yönetebilir. Linux İşletim Sisteminin faydalı verilerinin madenciliği ve analizi, bilgisayar adli analizinin önemli araçları ve araştırma yönü haline gelmiştir [11].

**3.1 Linux Adli Bilişimi**

Adli bilişim dünyasında, araştırmacıların aralarından seçim yapabilecekleri çeşitli araçlar ve seçenekler vardır. Bir bilgisayar araştırmacısının amacı, açık kaynaklı ve doğrulanabilir araçları kullanarak gerçeği ortaya çıkarmak olduğundan, araştırmacıyı ihtiyaç duyulan gerçeklerden uzak tutabilecek olası soyutlama katmanlarını ortadan kaldırır [19]. "Dcfldd" olarak bilinen böyle bir araç, adli bilişim uzmanlarının yararlanabileceği tüm özelliklere sahip DoD adli bilişim kütüphanesi tarafından geliştirilen Linux dd yardımcı programının değiştirilmiş bir sürümüdür [20]. Dosya sistemi analizi amacıyla kullanılan bu tür bir başka araç, araştırmacıların sürücü sektörlerini kolayca analiz etmesine ve karma kanıt dosyaları oluşturmasına olanak tanıyan açık kaynaklı Linux tabanlı bir araç seti olan "Sleuth Kit" ve "Autopsy" dir [21].

Genel olarak, bir adli bilişim uzmanının temel amacı, şüpheli cihazlardan kanıt şeklinde gerçeği ortaya çıkarmaktır. Tartışılabilecek sorun, kapalı kaynak programlarını kullanırken, bu tür bilgiler araçların kullanıcılarına açıklanmadığından, mevcut araç sürümüyle ilgili hata düzeltmelerini veya sorunları doğrulamanın çoğu zaman bir yolu olmamasıdır. Bu, kanıt toplama sürecinde çarpık sonuçlara veya çarpıklığa yol açabilir. Örneğin, Sleuth Kit gibi açık kaynaklı bir araç kullanmak, bir araştırmacı programın meşruiyetini ve birçok farklı şekilde nasıl çalıştığını doğrulayabilir, bunlardan biri hata düzeltmelerinin kamuya açık kaydı olabilir, başka bir seçenek kaynak kodun daha yeni sürümleriyle eskisini karşılaştırabilir. Üçüncü seçenek, kodun gerçek işlevini doğrulamaktır, dolayısıyla aracın nasıl olduğunu sunabilmek, kanıtı meşru bir şekilde ortaya çıkarır [19]. Bir kapalı kaynak aracı kullanırken araştırmacı, kendileri ve kanıt arasına soyutlama katmanları yerleştirir ve bu katmanların her biri ile kanıtın çarpıtılabileceği veya hatalara sahip olabileceği potansiyeli artar [10].

**3.2 Linux Sistemlerde Denetim ve Loglama**

Adli bilişim araştırmacıları, araştırmaları için herhangi bir yararlı bilgiyi toplamaya çalışmak için genellikle bir ana makinede bulunan denetim sistemlerine ve günlüklere bakarlar. Bu nedenle, bir araştırmacının, özellikle bir Linux sisteminde, denetleme ve günlüğe kaydetme ile ilgili kavram ve metodolojilere aşina olması önemlidir. Denetim, güvenlik amacıyla bir sistem veya uygulamada izlemek istediği eylemleri ifade eder. Loglama, bir uygulamada veya sistemde meydana gelen tüm etkinlikleri günlüğe kaydetmeyi ifade eder [22,23].

Her iki araç da (audit ve log) bir adli bilişim araştırmacısı için kanıtları ortaya çıkarma çabalarında veya bir şüphelinin bilgisayarında ne tür bir suçun işlendiğini daha iyi anlamaya yönelik girişimlerde çok yararlı olabilir. Linux, adli araştırmacılara yardımcı olabilecek pek çok yararlı günlüğe sahiptir ve Linux, varsayılan olarak çoğu dağıtımda yüklü olan ilginç bir denetim sistemine sahiptir.

Daha önce bahsedildiği gibi, Linux varsayılan olarak adli bilişim araştırmacıların Linux sistemlerini analiz ederken yararlanabilecekleri çok sayıda yararlı terminal komutuna ve günlüklere sahiptir. "history" ve "/usr/bin/w" gibi bazı basit komutlarla başlayalım. "history" komutu, oturum açmış mevcut kullanıcı tarafından yürütülen son komutları gösterir ve "usr/bin/w", şu anda kimin oturum açmış olduğunu ve her bir kullanıcının ne yaptığını görmeyi sağlar [8]. Bu komutlar, bir sistemde hemen ne olabileceğini anlamanın basit bir yoludur, ancak günlük dosyaları bir araştırmacı için çok daha yararlıdır çünkü genellikle şüpheli bir olayın meydana gelebileceği anı içerirler. Örneğin, "/var/log/faillog" dizini her başarısız kullanıcı oturum açma girişimini içerir ve bu, kötü bir aktörün bir sisteme girmeye çalışıp çalışmadığını belirlemeye çalışmak için yararlı bir günlük olabilir [24]. Diğer bir örnek, makineden yazdırılan tüm öğelerin kaydını gösteren "/var/log/lpr.log" dur. Bu, kurumsal casusluk davaları için harika bir araç olabilir ve hangi bilgilerin çalındığı iddiasını sağlamlaştırmak için mahkemede ek kanıtlar sağlayabilir. Bunlar, gerçekleştirilebilecek temel günlükler ve komutlardan yalnızca birkaçıdır; ancak, bilgisayar korsanlarının izlerini gizlemek için günlüklerin kolayca tehlikeye atılabileceğini ve düzenlenebileceğini anlamak önemlidir. Bu nedenle, gelişmiş bir denetim sistemine sahip olmak önemlidir.

Pek çok Linux dağıtımı varsayılan olarak, güvenlikle ilgili olayları izlemek için harika bir araç olan Linux Denetim sistemini (Linux audit log) kurmuştur [20,10].

Bir sistem çağrısı bu filtrelerden birinden geçtiğinde, dışlama filtresi aracılığıyla gönderilir, bu da denetim kuralı yapılandırmasına bağlı olarak onu daha fazla işlem için denetim arka plan programına gönderir.

Denetim sistemi, güvenlik göz önünde bulundurularak tasarlanmıştır ve bir kullanıcının günlük dosyalarındaki birçok yararlı bilgiyi denetlemesine ve kaydetmesine olanak tanır. Örneğin, tipik bir denetim sistemi günlük dosyasında şunlar bulunabilir. Söz konusu olayın sonucu ile birlikte olayın tarihi/saati ve türü, öznelerin ve nesnelerin hassasiyet etiketleri, olayı tetikleyen kullanıcının kimliği, herhangi bir güvenilir veri tabanındaki değişiklikler "/etc/passwd" gibi ve denetim yapılandırmasında yapılan değişiklikler ve Denetim günlük dosyalarına erişme girişimleri [10]. Bu, denetim sisteminin neler yapabileceğinin tam bir listesi değildir, ancak Linux Denetim sisteminden ne kadar yararlı bilginin elde edilebileceğini gösterir. Denetleme ve kayıt tutma, adli bilişim araştırmacıları için değerli bilgi araçlarıdır. Bir sistemde meydana gelen tüm olaylar ve söz konusu olayları tetikleyen kullanıcı hakkında bilgi içerirler. Linux bunun farkındadır ve güvenli denetim/günlük çözümleri sağlamak için birçok araç sağlamıştır [20,10].

**3.2.1 Linux Dosya Sistemi Günlük Dosyalarının Adli Bilişimi**

Dosya sistemi günlükleri, diskte bulunan verileri dosya sistemine gönderilmeden önce önbelleğe almak için kullanılır. Bu, güç kaybı veya sistem arızası olması durumunda hiçbir şeyin kaybolmamasını sağlamak için yapılır; ancak, böyle bir olay meydana gelirse, yazma işlemlerinin tamamlanmasına izin vermek için günlük yeniden oynatılabilir. Günlük analiz edilebilir ve üzerine yazılan veya silinen veri dosyalarını kurtarmak için kullanılabilir, çünkü önceki dosya yazma işlemleri genellikle sınırlı bir süre için günlükte saklanır. Linux'un ext2 sürümünden sonra, günlük tutma ReiserFS, ext3 ve ext4 gibi daha yeni dosya sistemlerine uygulanmıştır [20].

Günlük sistemi, bir işletim sisteminde adli bilişim için en önemli mekanizmadır. Linux sistemindeki adli öneme sahip olabilecek çeşitli kayıt mekanizmaları, [9]’da tartışılmaktadır.

Ext3 dosya sistemi, Linux uzantılı dosya sistemlerinin üçüncü varyasyonudur ve günlük kaydı özelliğini ilk uygulayan sistemdir. Ext3 süper blok yapısını kullanır, bu nedenle günlüğün boyutu yalnızca dosya sisteminin boyutuna bağlıdır. Günlükte oturum açılmış her girdi bir dönüşümlü sıraya konur, tüm girdilerin bir yorum bloğu ve bir tanımlayıcı bloğu vardır [20].

Dosya sistemi içindeki dosyalarda değişiklik yapıldığında, bunlar günlükte oturum açarlar ve bu değişiklikler bir veri kurtarma aracı kullanılarak keşfedilebilir. Ext3'te varsayılan günlük ayarlarını kullanırken, günlükte kaydedilen dosya sisteminde yalnızca meta verilere yapılan değişiklikler olur. Bu, bir dosya düzenlenirken tüm meta verilerin günlükte saklandığı anlamına gelir. Yalnızca meta verilerin günlüğe kaydedilmesi yerine, tüm veriler bağlama komutu seçeneği, “data = journal” kullanılarak günlüğe kaydedilebilir. Günlük girişlerini analiz etmek ve değiştirilmiş, silinmiş veya üzerine yazılmış dosyaları aramak için Java tabanlı bir veri kurtarma aracı kullanılabilir. Veri kurtarma aracı, komut satırı bağımsız değişkeni olarak iletilen bir dosya sistemini açmak için bir “FileInputStream” nesnesi kullanır ve dosyayı bayt seviyesinde okur. Dosyayı açtıktan sonra, bilgiler süper bloktan çıkarılır ve ileride kullanılmak üzere saklanır [25,26].

Günlük dosyalarının Linux işletim sisteminin ext3 sürümüne uygulanması, verilerin tek bir yerde depolanmasına yardımcı olmanın harika bir yoludur. Tüm verilerin tek bir yerde saklanmasıyla birlikte, bu, bir veri kurtarma aracının günlüklerde kullanımını çok daha kolay hale getirir. Veri kurtarma araçları, günlük bloklarında değişiklikler, silmeler ve üzerine yazılan dosyalar için arama yapmak için kullanılabilir. Araç, her bloktan ne tür verilerin kurtarıldığını ve günlük bloğundaki bir dosyada yapılan düzenleme türüne göre nasıl kurtarılacağını belirlemek için birçok adım atar [20].

**3.2.2 Linux Dosya Sistemi Yapısı**

Linux işletim sisteminin en önemli bileşenlerinden biri olan dosya sistemi, sadece verileri değil aynı zamanda yapıyı da içeren bir veri koleksiyonu olan dokümanı yönetmek ve depolamak için kullanılır. Linux işletim sisteminin temel dosya sistemi olan Ext2, kullanılan ekipmanın mantık bölümlerini, tüm dosya sistemi hakkında bazı tanımlayıcılar ve veri blokları içeren birçok veri bloğu grubuna ayırır [20].

**3.3 Linux Adli Bilişiminde Kullanılan Metotlar**

Linux tabanlı web sunucuları sürekli saldırı altındadır. SophosLabs'ta, 2013 yılında bir günde ortalama 16.000-24.000 web sitesinin güvenliği ihlal edilmiştir [27]. Linux sistemleri gerçekten de kötü amaçlı yazılımlar tarafından saldırıya uğrar. Microsoft'un işletim sistemi tasarımı, belgeleri yürütülebilir yükleri yükleyebilen bazı özellikler içerir. Yazılım kancalarından ve kod saplamalarından (kayıt defteri) oluşan bir veri tabanının kullanılması da işleri basitleştirmiştir [27,28].

Linux kötü amaçlı yazılımları, Windows virüslerine kıyasla ne yaptığından ve nasıl yaptığından oldukça farklıdır. Önemli işletim sistemi dizinleri, kötü amaçlı yazılım tarafından bilgisayar sistemini bir bütün olarak etkilemek için kullanılabilir. Ek olarak, içerideki kötü niyetli kişilerin riski her zaman vardır. Linux sistemlerine yönelik saldırılar, web tarayıcıları veya Java kapsayıcıları gibi sistem hizmetlerindeki hataları istismar etmeyi amaçlama eğilimindedir. Bunlar da sıklıkla yükseltilmiş ayrıcalıklarla çalışmazlar, bu nedenle genellikle hedeflenen hizmetin davranışını değiştirmek ve muhtemelen devre dışı bırakmak için bir istismar bulunur. Kötü amaçlı yazılım, Linux dosya sistemindeki çeşitli dizinleri kullanarak onu bir hizmet olarak çalışacak ve Bilgisayara zarar verecek şekilde yerleştirir. Ayrıca, içerideki kötü niyetli kişilerin faaliyetleri de dosya sisteminde depolanır. Bu, sistemdeki kötü niyetli faaliyetlerin izlerini bulmak için Linux dosya sistemi altındaki dizinlerin adli incelemesini yapma ihtiyacını ortaya çıkarır [27].

Ubuntu işletim sisteminin bir dosya sistemi, adli öneme sahip birçok yapılandırma bilgisini ve bilgiyi koruyabilmekte ve yönetebilmektedir. Ubuntu işletim sisteminin faydalı verilerinin madenciliği ve analizi, bilgisayar sistemine yapılan saldırının artmasıyla birlikte zorunlu hale geldi. Dosya Sisteminin araştırılması, vakayla ilgili bilgilerin toplanmasına yardımcı olabilir [28]. Ubuntu işletim sistemi, Linux işletim sisteminin dağıtımlarından biridir. Ubuntu çekirdeklerinin çoğu, varsayılan Linux çekirdeğidir. Ubuntu, genellikle bir ağaç yapısı olarak kabul edilen Linux dosya sistemini kullanır. Ubuntu, Ext4'ü varsayılan dosya sistemi olarak kullanıyor. Ext4, daha önce varsayılan dosya sistemi olan Ext3'ün bir evrimidir. Ext dosya sisteminin gelişimi Tablo 4'de özetlenmiştir. Linux bilgisayarlar, bilgisayar korsanlarının saldırısına çok açıktır. Linux kutuları, esasen merkezi bir kontrol noktası için genellikle sunucu olarak kullanılır. Aslında, bilgisayar korsanları tarafından honeypot'lara indirilen kötü amaçlı yazılımların yaklaşık %70'i Linux/Rst-B ile enfekte olmuştur. EXT dosya sistemine ait özelliklerin bulunduğu bilgiler Tablo 4’da gösterilmektedir [27,28].

**Tablo 4** EXT Ailesi Özellikleri ve Sınırlamaları

|  |  |  |  |
| --- | --- | --- | --- |
| **Linux Dosya Sistemi** | **Yayın Tarihi** | **Özellikleri** | **Sınırlar** |
| **EXT** | 1992 | Kullanılan Sanal Dosya sistemi kavramı | Dosya erişimi için ayrı zaman damgası desteği yok |
| **EXT2** | 1993 | Dosya Sıkıştırma eklendi | Günlük kaydı özelliği yok |
| **EXT3** | 1999 | Günlük kaydı eklendi, çevrimiçi dosya sistemi büyümesi | Kapsamlar, dosya numaralarının dinamik tahsisi ve blok alt tahsisi gibi eksik özellik |
| **EXT4** | 2006 | Kapsam tabanlı depolama, EXT2 ve EXT3 ile geriye dönük uyumluluk, Çevrimiçi birleştirme | Güvenlik sorununa neden olan silindikten sonra dosyanın üzerine yazmayın. |

Ubuntu Linux ve Free BSD'deki çeşitli dosya sistemlerinin karşılaştırmalı bir çalışması da [29]'de yapılmıştır. Bilgisayar adli bilimi için Linux dosya sistemi analizi uygulamaları talebini karşılamak için, [30] 'de Linux dosya sistemini analiz etmek için nesne yönelimli bir yöntem önerilmiştir. Makale ayrıca, sınıflar arasındaki miras ilişkisi ve sınıfın kapsüllenmesi ile farklı veri kaynaklarını derinlemesine analiz etti ve Linux dosyası bilgilerini kullanıcılara dostça bir ara yüzde gösterdi. Linux işletim sistemi, dünya çapında birçok iş hizmetinde bir sunucu sistemi olarak kullanılmıştır. Bir sunucudaki yetkisiz izinsiz girişler geometrik bir ilerleme ile sürekli olarak artmaktadır. Tersine, izinsiz giriş kazalarına karşı koruma ve önleme teknikleri kesinlikle yetersizdir. Dijital adli bilişim araştırmasında tehlikeye atılmış bir Linux sistemiyle başa çıkmak için yeni bir çerçeve geliştirildi ve uygulandı [31]. Linux Forensics ve Linux sisteminin adli araştırması için çeşitli adli bilişim ve olay müdahale süreçlerinde kullanılan araçlarla ilgili sorunlar [32] 'de tartışılmıştır.

**4. Kullanılan Yöntem ve Önerilen Yaklaşım**

Linux İşletim Sistemlerinde, sistem üzerinde gerçekleştirilen işlemlerle ilgili bilgiler dosya sisteminde tutulur. Adli bilişim araştırmacısının sistemde gerçekleştirilen çeşitli faaliyetlerin potansiyel dijital kanıtlarını bulmasına yardımcı olabilecek dosya sistemindeki bazı dosyalar ve dizinlerdir ve bunlar aşağıda detaylı olarak verilmiştir. Linux Dosya Sisteminin temel olarak incelenmesi gereken alanları aşağıdaki gibidir.

* /etc/rc.d
* /etc/init.d
* /etc/NetworkManager/System-connections
* /etc/passwd
* /etc/shadow
* /etc/profile
* /etc/networks
* /etc/hosts
* /etc/cron.d,
* /etc/cron.daily
* /etc/cron.weekly
* /etc/cron.d
* /etc/cron.daily
* /usr/bin
* /usr/lib
* /usr/local/share/recently-used.xbel
* /var/log/syslog
* /var/log/lastlog
* /var/log/faillog
* /var/tmp/
* /dev
* /proc/net/netstat
* /proc/net/dev\_mcast
* /proc/cpuinfo
* /proc/PID/exe

**4.1 Linux Sistemlerden Veri Elde Etme Yöntemi**

Bu makalede yapılan tüm araştırmalar ve edinilen bilgiler doğrulusunda varılan kazanımların ve kullanılan araçların olay müdahale süreçleri içerisinde istenilen hızı kazandırmadığı görülmüştür. Bu sebeple Linux sistemlerden dosya elde etmek için bash programlama dilinde aşağıda açıklaması yapılan araç geliştirilmiştir. İlgili araç henüz tüm Linux dağıtımlarında istikrarlı çalışmasa dahi yukarıda açıklamaları verilen verileri çok hızlı şekilde toplayabilmesi ile bu makalede ele alınan soruna çözüm olabilmektedir. Geliştirilen aracın nasıl çalıştığı ve hangi verileri topladığı aşağıda açıklanmıştır.

Önerilen yöntem, bash script dilinde yazılmıştır. Yöntemin temel amacı Linux işletim sistemi üzerinde yerleşik olarak geliştirilmiş araçların kullanarak sistem verilerini toplamaktır. Toplanan verileri makine ismini de ekleyerek bir zip dosyasına kaydetmektedir. Uygulamayı çalıştırmak için yönetici haklarına sahip olmak gereklidir. Uygulamanın temelini standart linux komutları oluşturmaktadır. Yukarıda Linux dosya sisteminde incelenmesi gereken alanların dosya yolları verilmiştir.

Uygulama temel olarak sistem komutlarını sırasıyla çalıştırıp çıktıları bir konuma yazmaktadır. Buradaki çıktılar adli bilişim açısından önem arz eden verileri içermektedir. Bu çıktıları okuyan analistler yaşanan siber güvenlik vakasına çözüm üretirler. Uygulamada çalışan fonksiyonlara bir örnek Şekil 1’de verilmiştir.

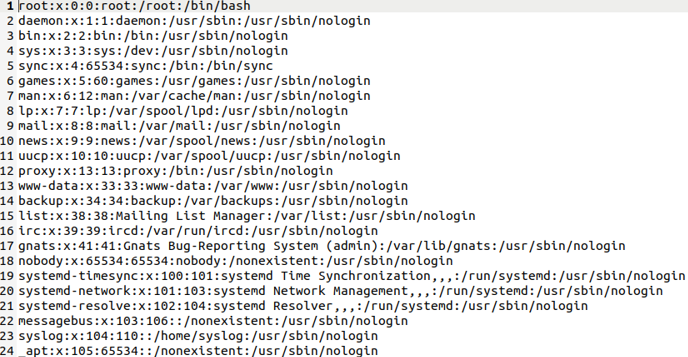


**Şekil 1** Çalışan Servislerin Tespiti İçin Gerekli fonksiyon

Geliştirilen araç bu alanlardaki verilerin okunmasını sağlamakta ve bir metin çıktısı oluşturmaktadır. Aynı zamanda bu dosyaların özet fonksiyon değerlerinin de hesaplayarak veri bütünlüğünü de sağlamaktadır. Bu işlemi tek seferde ve hızlıca yapması en temel kazanımıdır. Normal şartlarda bu verilerin elde edilebilmesi için ya sisteme uzak bağlantı sağlanarak tek tek ilgili alanlardan verilerin okunması gerekir ya da sistemin imajının alınması ve Bölüm 3’te bahsedilen Linux Adli Bilişim araçlarının kullanılması gerekmektedir. Bizim yöntemimiz sayesinde buradaki lisans maliyeti ve zamandan kazanım sağlanmaktadır. Şekil 2 ve 3’de araca ait örnek çıktılar gösterilmiştir.



**Şekil 2** Özet Fonksiyonlarının Hesaplanması



**Şekil 3** passwd dosya içeriğinin görüntülenmesi

**5. Bulgular ve Çıkarımlar**

Literatür taraması ve araştırmalar sonucunda, olay müdahale planının bilgi güvenliği tehditlerinin tespit edilmesi ve ortadan kaldırılması için çok kritik olduğu görülmüştür. Olay müdahalesi, teknik sorunlar dışında birçok yönü içerdiği anlaşılmıştır. Bir olay müdahale ekibinin dikkate alması gereken yönetimsel, yasal ve sosyal sorunlarının olduğu görülmüştür. Bu sorunların temelinde incelemenin pratik ve hızlı bir şekilde yapılması gerektiği değerlendirilmiştir. Bu sorunların başında gelen Linux sistemlerin incelenmesi yönüne çözüm üretilmiştir. Üretilen çözümde zamandan ve maliyetten kazanım mevcuttur. Önerilen script sayesinde kurum içerisine var olan bir uzak yönetim çözümü ile Linux sistemlerinin analizi oldukça kolay hale gelmektedir.

**6. Değerlendirme ve Sonuç**

Yapılan çalışmalar sonucunda önerilen çözümün adli bilişim ve olay müdahalesi süreçlerinde katkı sağladığı gözlemlenmiştir. Kurumların bu çözüm sayesinde olay müdahale süreçlerini daha efektif yönetebileceği değerlendirilmektedir. Ayrıca bu çözüm yöntemi sektörde bu alanda çalışan araştırmacılar tarafından da kullanıldığı ve başarılı sonuçlar elde edildiği gözlemlenmiştir.

**Kaynaklar**

[1] Chen J, Su C, Yeh K-H, Yung M (2018) Special issue on advanced persistent threat. vol 79. Elsevier,

[2] Niakanlahiji A, Wei J, Chu B-T A natural language processing based trend analysis of advanced persistent threat techniques. In: 2018 IEEE International Conference on Big Data (Big Data), 2018. IEEE, pp 2995-3000

[3] Ahmad A, Webb J, Desouza KC, Boorman J (2019) Strategically-motivated advanced persistent threat: Definition, process, tactics and a disinformation model of counterattack. Computers & Security 86:402-418

[4] Auty M (2015) Anatomy of an advanced persistent threat. Network Security 2015 (4):13-16

[5] Hekim H, BAŞIBÜYÜK O (2013) Sİber Suçlar ve Türkİye’Nİn Sİber Güvenlİk Polİtİkalari. Uluslararası Güvenlik ve Terörizm Dergisi 4 (2):135-158

[6] Karantzas G, Patsakis C (2021) An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors. Journal of Cybersecurity and Privacy 1 (3):387-421

[7] Javed AR, Ahmed W, Alazab M, Jalil Z, Kifayat K, Gadekallu TR (2022) A Comprehensive Survey on Computer Forensics: State-of-the-art, Tools, Techniques, Challenges, and Future Directions. IEEE Access

[8] Garfinkel SL (2010) Digital forensics research: The next 10 years. digital investigation 7:S64-S73

[9] Ling T The study of Computer forensics on Linux. In: 2013 International Conference on Computational and Information Sciences, 2013. IEEE, pp 294-297

[10] Reddy N (2019) Linux forensics. In: Practical Cyber Forensics. Springer, pp 69-100

[11] Nikkel B (2021) Practical Linux Forensics: A Guide for Digital Investigators. no starch Press,

[12] Reith M, Carr C, Gunsch G (2002) An examination of digital forensic models. International Journal of Digital Evidence 1 (3):1-12

[13] Henkoğlu T (2020) Adli bilişim: Dijital delillerin elde edilmesi ve analizi. Pusula,

[14] Schneier B (2014) The future of incident response. IEEE Security & Privacy 12 (5):96-96

[15] Casey E (2009) Handbook of digital forensics and investigation. Academic Press,

[16] Sabillon R (2022) Cybersecurity Incident Response and Management. In: Research Anthology on Business Aspects of Cybersecurity. IGI Global, pp 611-620

[17] Andrade R, Torres J, Cadena S Cognitive security for incident management process. In: International Conference on Information Technology & Systems, 2019. Springer, pp 612-621

[18] Thompson EC (2018) Cybersecurity incident response: How to contain, eradicate, and recover from incidents. Apress,

[19] Altheide C, Carvey H (2011) Digital forensics with open source tools. Elsevier,

[20] Amarchand G, Munn K, Renicker S A Study on Linux Forensics.

[21] Carrier B (2005) File system forensic analysis. Addison-Wesley Professional,

[22] Clarke GE (2018) CompTIA security+ certification study guide (exam SY0-501). McGraw-Hill Education,

[23] Jones KJ, Bejtlich R, Rose CW (2005) Real digital forensics: computer security and incident response. Addison-Wesley Professional,

[24] Easttom C (2017) System forensics, investigation, and response. Jones & Bartlett Learning,

[25] Sachowski J (2018) Digital Forensics and Investigations: People, Processes, and Technologies to Defend the Enterprise. CRC Press,

[26] Leigland R, Krings AW (2004) A formalization of digital forensics. International Journal of Digital Evidence 3 (2):1-32

[27] Patil DN, Meshram BB (2016) Digital forensic analysis of ubuntu file system. Int J Cyber Secur Digit Forensics 4 (5):175-186

[28] Patil DN, Meshram BB An Evidence Collection and Analysis of Ubuntu File System.

[29] Yang K-p, Wallace K (2011) File Systems in Linux and FreeBSD: A Comparative Study. Journal of Emerging Trends in Computing and Information Sciences 2 (9)

[30] Chen W, Liu C-m The analysis and design of Linux file system based on computer forensic. In: 2010 International Conference On Computer Design and Applications, 2010. IEEE, pp V2-60-V62-64

[31] Choi J, Savoldi A, Gubian P, Lee S, Lee S Live forensic analysis of a compromised linux system using LECT (Linux Evidence Collection Tool). In: 2008 International Conference on Information Security and Assurance (isa 2008), 2008. IEEE, pp 231-236

[32] Grundy B (2014) Advanced artifact analysis. European Union Agency for Network and Information Security