

ADA LOVELACE AKADEMİ

Siber Güvenlik Final Ödevi

-LOG ANALİZİ-

03/09/2024

MAKBULE ARI

İÇİNDEKİLER

1) Log Nedir?.....	2
1.1) Logların Önemi.....	2
1.2) Log Türleri	2
1.3) Log Özellikleri	3
1.4) Analiz Gerekliliği	4
2) acsess.log Analizi	5
3) acsess.log.1 Analizi	15
4) acsess.log.2 Analizi	25
5) acsess.log.3 Analizi	37
Kaynakça	53

1) Log Nedir?

Log, bir sistem, uygulama veya cihaz tarafından üretilen ve çeşitli olayları, işlemleri veya durumları zaman damgasıyla kaydeden bir veri dosyasıdır.

1.1) Logların Önemi

Log kayıtları sisteme giriş-çıkış işlemleri, sistem içerisinde yapılan değişiklikler, sistem içinde başarısız oturum açma işlemleri, sisteme hangi kullanıcıların girdiğine dair tutulan bu kayıtlar herhangi bir siber saldırı veya güvenlik sorunlarında geriye yönelik analiz yapmamızı kolaylaştıracaktır.

1.2) Log Türleri

Loglar, kullanım amacına ve içeriği bilgilere göre çeşitli türlere ayrılabilir. Bazı yaygın log türleri:

- **Erişim Logları (Access Logs):** Kullanıcıların bir sisteme veya uygulamaya erişimlerini kaydeder. Web sunucularında, hangi sayfaların ziyaret edildiğini, hangi kullanıcıların giriş yaptığını gösterir.
- **Hata Logları (Error Logs):** Sistem veya uygulamalarda meydana gelen hataları ve sorunları kaydeder. Hata kodları, mesajlar ve hatanın olduğu yer hakkında bilgi içerir.
- **Uygulama Logları (Application Logs):** Yazılım uygulamalarının çalışma sürecindeki olayları kaydeder. Kullanıcı etkileşimi, işlem sonuçları ve uygulama içi durum değişikliklerini içerir.
- **Sistem Logları (System Logs):** İşletim sistemi tarafından üretilir ve sistemin genel durumu, hizmetlerin durumu ve donanım bilgileri hakkında bilgi verir.
- **Güvenlik Logları (Security Logs):** Güvenlik olaylarını ve potansiyel tehditleri izler. Yetkisiz erişim girişimleri, kullanıcı aktiviteleri ve güvenlik ihlalleri hakkında bilgi içerir.
- **Veritabanı Logları (Database Logs):** Veritabanı yönetim sistemleri tarafından üretilir ve veri sorguları, işlemler, hata durumları ve performans verilerini içerir.
- **İşlem Logları (Transaction Logs):** Özellikle finansal işlemler veya veri tabanı işlemleri gibi belirli bir işlemle ilgili ayrıntıları kaydeder. İşlem kimlikleri ve veri değişikliklerini içerir.
- **Ağ Logları (Network Logs):** Ağ cihazlarından gelen verileri içerir. Ağ trafigi, bağlantı sorunları ve veri paketlerinin durumunu kaydeder.

- **İş Günlükleri (Job Logs):** Arka plan işlemleri veya zamanlanmış görevlerin çalışma durumunu izler. İşin başlangıç ve bitiş zamanları, başarı durumu veya hata bilgilerini içerir.
- **Performans Logları (Performance Logs):** Sistem veya uygulamanın performans metriklerini kaydeder. CPU kullanımı, bellek tüketimi ve yanıt süreleri gibi bilgileri içerir.

Bu log türleri; sistemlerin ve uygulamaların yönetilmesini, sorunların çözülmesini ve güvenliğin sağlanması kolaylaştırır.

1.3) Log Özellikleri

Log dosyaları genellikle olayların zaman damgaları, olay türleri, açıklamalar ve kaynak bilgileri gibi ayrıntıları içerir.

- ✓ **Zaman Damgası:** Her log kaydında olayın tam tarih ve saatı belirtilir. Bu, olayların zaman sırasını takip etmek ve geçmişteki olayları kronolojik sırada incelemek için önemlidir.
- ✓ **Olay Türü:** Log kayıtları, genellikle olayın türünü belirtir. Bu türler arasında bilgi mesajları, uyarılar, hatalar ve kritik durumlar yer alabilir.
- ✓ **Olay Açıklaması:** Olayın veya işlemin detaylı bir açıklaması bulunur. Bu açıklama, olayın ne olduğunu ve sistemde ne tür bir etki yarattığını açıklar.
- ✓ **Kaynak Bilgisi:** Olayın meydana geldiği sistem, uygulama veya bileşen hakkında bilgi verir. Bu, olayın hangi yazılım veya donanım bileşeni tarafından üretildiğini belirler.
- ✓ **Hata Kodu veya Durum Kodu:** Özellikle hata durumlarında, olayın nedeni veya durumu hakkında bilgi veren bir kod veya mesaj içerir. Bu, hata ayıklama ve sorun çözme süreçlerinde yardımcı olur.
- ✓ **Kullanıcı Bilgisi:** Olayı gerçekleştiren veya etkilenen kullanıcı hakkında bilgi verir. Bu, kullanıcı adı veya kimlik bilgilerini içerebilir.
- ✓ **Log Formatı:** Loglar, genellikle düz metin formatında saklanır, ancak bazı sistemler özel veri formatları kullanabilir. Log formatı, verinin nasıl yapılandırıldığını ve okunabilirliğini etkiler.
- ✓ **Saklama Süresi:** Log dosyalarının saklama süresi, sistem yapılandırmasına ve organizasyonun ihtiyaçlarına göre değişir. Loglar belirli bir süre sonra otomatik olarak temizlenebilir veya arşivlenebilir.
- ✓ **Erişim ve Güvenlik:** Log dosyalarına erişim, genellikle yalnızca yetkili kullanıcılarla sınırlıdır. Logların güvenliği, yetkisiz erişimlerin önlenmesi ve veri bütünlüğünün korunması açısından önemlidir.

- ✓ **Analiz ve Raporlama:** Loglar, genellikle analiz araçları ve yazılımları kullanılarak incelenir. Bu araçlar, log verilerini özetler, grafikler oluşturur ve raporlar sunar.
- ✓ **Veri Bütünlüğü:** Logların değiştirilmemesi ve tam ve doğru bir şekilde saklanması gereklidir. Veri bütünlüğü, logların güvenilirliğini sağlar.

1.4) Analiz Gerekliliği

Log analizi, sistemin sağlıklı çalışmasını sağlamak, güvenlik açılarını tespit etmek ve performans sorunlarını çözmek için düzenli olarak yapılmalıdır.

Bu rapor, final ödevi kapsamında gerçekleştirilen kapsamlı log analizinin sonuçlarını ve bulgularını özetlemektedir.

2) access.log Analizi

Bu log dosyası, bir web sunucusunun erişim kayıtlarını içeriyor. Her bir kayıt istemcinin IP adresi, zaman damgası, yapılan istek (HTTP metod ve yol), sunucunun verdiği yanıt durumu kodu, yanıt boyutu, yönlendiren URL ve kullanıcı aracı bilgilerini içermektedir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log | wc -l
405
```

Resim 1: cat access.log | wc -l

cat access.log | wc -l

Bu komut, access.log dosyasındaki toplam satır sayısını belirler. cat komutu dosyanın içeriğini ekrana yazdırırken, wc -l komutu bu içeriği satır sayısına dönüştürür.

Analiz: Dosyanın 405 satır içeriği tespit edilmiştir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log | grep "ERROR"
```

Resim 2: cat access.log | grep "ERROR"

Log dosyasında 'ERROR' kelimesini içeren satırları bulmak amacıyla grep "ERROR" komutu kullanılmıştır. Grep komutu, bir dosya içindeki metinleri belirli bir desenle aramak için kullanılır ve aranan desenle eşleşen satırları ekrana yazdırır.

Analiz: Yapılan aramada 'ERROR' kelimesini içeren herhangi bir satır tespit edilmemiştir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log | cut -d " " -f 1 | sort | uniq -c | sort -nr
 186 146.241.73.240
   69 193.230.127.66
   45 68.160.80.187
   42 36.40.116.182
   39 61.14.246.6
   25 66.48.84.188
```

Resim 3: `cat access.log | cut -d '' -f 1 | sort | uniq -c | sort -nr`

```
cat access.log | cut -d ' ' -f 1 | sort | uniq -c | sort -nr
```

Komutu kullanılarak access.log dosyasındaki IP adresleri sıklıklarına göre azalan şekilde sıralanmış ve her bir IP adresinin tekrar sayısı belirlenmiştir.

Analiz: Sonuç olarak, analiz edilen verilere göre 146.241.73[.]240 IP adresi, toplamda 186 kez erişim sağlanarak en sık karşılaşılan IP adresi olmuştur.

Resim 4: cat access.log | cut -d " " -f 1,7,9,10 | grep -i "login"

```
cat access.log | cut -d " " -f 1,7,9,10 | grep -i "login"
```

Bu komut, acsess.log dosyasındaki her satırdan belirli sütunları çıkarır ve bu sütunlarda "**login**" terimini büyük/küçük harf duyarlılığı olmadan arar.

Log kayıtları incelediğinde aşağıdaki bulgular tespit edilmiştir:

Yanıt Boyutları:

- **947 Byte:** Çoğu yanıt bu boyutta olup, standart bir içerik miktarını göstermektedir.
- **919 Byte:** Bu boyut, içerikte küçük bir değişiklik veya farklı bir yanıtın varlığını gösterebilir.
- **0 Byte:** Yanıtın hiç içerik içermediğini veya yanıtın eksik olduğunu işaret eder. Bu durum, bir hata veya sorun olduğunu gösterebilir.

HTTP Yanıt Kodları:

- **200:** Çoğu yanıt başarılı bir şekilde gerçekleşmiştir.
- **302:** En son yanıt kodu, isteğin başka bir URL'ye yönlendirilmiş olduğunu belirtir. Bu genellikle geçici bir yönlendirmeyi işaret eder ve belirli bir URL'ye erişimde bir değişiklik veya yönlendirme yapılmışının değiştiğini gösterir.

Potansiyel Saldırı Göstergeleri:

- **0 Byte Yanıtlar:** Yanıt boyutunun 0 byte olması, genellikle anormal bir durumu işaret edebilir. Bu durum, sunucunun düzgün çalışmadığını veya veri iletimi sırasında bir sorun yaşandığını gösterebilir. Saldırganlar, sunucunun işleyişini bozmak veya veriyi eksik iletme amacıyla bu tür yanıtları kullanabilirler.
- **HTTP 302 Yanıt Kodu:** HTTP 302 yönlendirme kodu, genellikle geçici bir yönlendirme olduğunu belirtir. Ancak, bu yönlendirme kötü niyetli sitelere yönlendirme amacıyla veya kullanıcıları yanılmak için kullanılabilir. Bu kod tek başına bir saldırıyı kesin olarak işaretlemez, ancak yönlendirme hedeflerinin ve amaçlarının dikkatle incelenmesi gereklidir.

Analiz: Sonuç olarak log kayıtlarında, diğer satırlardan farklı olarak ilk satırdaki yanıt boyutunun 919 byte ve son satırdaki yanıt boyutunun 0 byte olduğu tespit edilmiştir. Ayrıca, çoğu yanıtın 200 HTTP kodu ile döndüğü, ancak en son yanıtın 302 HTTP kodu ile döndüğü belirlenmiştir. Yanıt boyutlarındaki ve HTTP kodlarındaki bu anormallikler, potansiyel sorunlar veya saldırı göstergeleri olarak değerlendirilmektedir. Bu durumların daha detaylı analiz edilmesi ve gerekli önlemlerin alınması önem arz etmektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log | grep "146.241.73.240" | grep "POST /login" | wc -l
100
```

Resim 5: cat access.log | grep "146.241.73.240" | grep "POST /login" | wc -l

cat access.log | grep "146.241.73.240" | grep "POST /login" | wc -l

Bu komut, belirli bir IP adresinden gelen "POST /login" isteklerinin sayısını hesaplar. İlgili IP adresine ait "POST /login" isteklerini filtreleyerek, bu tür isteklerin toplam sayısını belirler.

Analiz: Analiz sonucuna göre 146.241.73[.]240 IP adresi üzerinden 100 kez "POST /login" isteği yapılmıştır. Bu kadar sık tekrar eden giriş denemeleri, brute-force saldırısı ihtimalini artırmaktadır. Giriş denemelerinin bu sayıda olması, şüpheli bir aktivite olarak değerlendirilmelidir. Bu duruma karşı, söz konusu IP adresinin engellenmesi veya daha detaylı bir inceleme yapılması önerilmektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log | cut -d " " -f 1,4,7,9,10
66.48.84.188 [26/Apr/2023:21:33:38 / 200 2134
66.48.84.188 [26/Apr/2023:21:33:38 /resources/static/js/header.js 200 368
66.48.84.188 [26/Apr/2023:21:33:38 /resources/static/css/footer.css 200 1240
66.48.84.188 [26/Apr/2023:21:33:38 /resources/css/XsBlog.css 200 4488
66.48.84.188 [26/Apr/2023:21:33:38 /resources/images/blog.svg 200 2666
66.48.84.188 [26/Apr/2023:21:33:38 /image/blog/posts/4.jpg 200 41222
66.48.84.188 [26/Apr/2023:21:33:38 /image/blog/posts/39.jpg 200 110838
66.48.84.188 [26/Apr/2023:21:33:39 /footer 404 31
66.48.84.188 [26/Apr/2023:21:33:39 /resources/static/images/logoa.svg 200 3097
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/17.jpg 200 33543
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/63.jpg 200 141236
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/42.jpg 200 203200
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/46.jpg 200 253600
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/66.jpg 200 256494
66.48.84.188 [26/Apr/2023:21:33:39 /resources/static/images/ps-X-a.svg 200 379
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/26.jpg 200 29354
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/45.jpg 200 135001
66.48.84.188 [26/Apr/2023:21:33:39 /image/blog/posts/53.jpg 200 192118
66.48.84.188 [26/Apr/2023:21:33:40 /favicon.ico 200 1654
66.48.84.188 [26/Apr/2023:21:33:44 /post?postId=5 200 2623
66.48.84.188 [26/Apr/2023:21:33:44 /resources/images/avatarDefault.svg 200 3372
66.48.84.188 [26/Apr/2023:21:33:44 /footer 404 31
66.48.84.188 [26/Apr/2023:21:33:48 /post?postId=4 200 2785
66.48.84.188 [26/Apr/2023:21:33:48 /footer 404 31
66.48.84.188 [26/Apr/2023:21:33:51 /post?postId=9 200 2855
61.14.246.6 [26/Apr/2023:21:33:51 /footer 404 31
61.14.246.6 [26/Apr/2023:21:33:54 /post?postId=7 200 3061
61.14.246.6 [26/Apr/2023:21:33:54 /footer 404 31
61.14.246.6 [24/Apr/2023:23:31:17 /post/comment 302 0
61.14.246.6 [24/Apr/2023:23:31:17 /post/comment/confirmation?postId=7 200 880
61.14.246.6 [24/Apr/2023:23:31:17 /footer 404 31
61.14.246.6 [24/Apr/2023:23:31:19 /post?postId=7 200 3096
61.14.246.6 [24/Apr/2023:23:31:19 /footer 404 31
61.14.246.6 [24/Apr/2023:23:31:44 / 200 2134
```

Resim 6: cat access.log | cut -d " " -f 1,4,7,9,10

```

61.14.246.6 [24/Apr/2023:23:31:44 /resources/static/css/footer.css 200 1240
61.14.246.6 [24/Apr/2023:23:31:44 /resources/css/XsBlog.css 200 4488
61.14.246.6 [24/Apr/2023:23:31:44 /resources/static/js/header.js 200 368
61.14.246.6 [24/Apr/2023:23:31:44 /resources/images/blog.svg 200 2666
61.14.246.6 [24/Apr/2023:23:31:44 /image/blog/posts/4.jpg 200 41222
61.14.246.6 [24/Apr/2023:23:31:44 /image/blog/posts/39.jpg 200 110838
61.14.246.6 [24/Apr/2023:23:31:44 /footer 404 31
61.14.246.6 [24/Apr/2023:23:31:44 /image/blog/posts/17.jpg 200 33543
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/66.jpg 200 256494
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/26.jpg 200 29354
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/63.jpg 200 141236
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/46.jpg 200 253600
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/42.jpg 200 203200
61.14.246.6 [24/Apr/2023:23:31:45 /resources/static/images/ps-X-a.svg 200 379
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/45.jpg 200 135001
61.14.246.6 [24/Apr/2023:23:31:45 /resources/static/images/logoa.svg 200 3097
61.14.246.6 [24/Apr/2023:23:31:45 /image/blog/posts/53.jpg 200 192118

```

Resim 7: cat acsess.log | cut -d “ “ -f 1,4,7,9,10

cat acsess.log | cut -d “ “ -f 1,4,7,9,10

Bu komut, `acsess.log` dosyasındaki her satırda yalnızca 1., 4., 7., 9. ve 10. sütunları seçer ve görüntüler.

- **1. Sütun:** IP adresi - İsteği gönderen cihazın IP adresini belirtir.
- **4. Sütun:** Tarih ve saat - İsteğin yapıldığı tarih ve saati gösterir.

Log verilerinde olayların zaman içindeki dağılımını anlamak, hataları tespit etmek, performansı izlemek, güvenlik olaylarını analiz etmek ve kullanıcı davranışlarını takip etmek için kritik öneme sahiptir. Bu bilgiler, olayların sıralamasını ve etkilerini değerlendirmede yardımcı olur, etkili sorun çözme ve analiz için temel sağlar.

- **7. Sütun:** İstenilen kaynak yolu - HTTP isteği yapılan URL yolunu belirtir. Bu, istek yapılan sayfanın veya dosyanın adresini gösterir.
- **9. Sütun:** HTTP yanıt kodu - Sunucunun isteğe verdiği yanıtın durum kodunu gösterir,
- **10. Sütun:** Yanıt boyutu - Sunucunun yanıtının byte cinsinden boyutunu belirtir.

Analiz: Log kayıtlarından görüldüğü üzere, aynı saniye içerisinde çok fazla işlem yapılmaktadır. Bu durum, yüksek işlem hacmini veya potansiyel bir sorun olduğunu işaret edebilir ve daha detaylı bir inceleme gerektirebilir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log | cut -d " " -f 1,4,5,6,7,8,9,10,11,12,13,14,15,16,17 | grep "404"
66.48.84.188 [26/Apr/2023:21:33:39 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
66.48.84.188 [26/Apr/2023:21:33:44 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
66.48.84.188 [26/Apr/2023:21:33:48 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
61.14.246.6 [26/Apr/2023:21:33:51 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
61.14.246.6 [26/Apr/2023:21:33:54 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
61.14.246.6 [24/Apr/2023:23:31:17 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
61.14.246.6 [24/Apr/2023:23:31:19 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
61.14.246.6 [24/Apr/2023:23:31:44 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
61.14.246.6 [24/Apr/2023:23:31:51 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
61.14.246.6 [24/Apr/2023:23:54:01 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
36.40.116.182 [24/Apr/2023:23:54:52 +0000] "GET /post?postId=11 HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X
36.40.116.182 [24/Apr/2023:23:55:17 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
193.230.127.66 [25/Apr/2023:23:27:51 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
146.241.73.240 [25/Apr/2023:23:29:54 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
146.241.73.240 [26/Apr/2023:21:40:11 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
```

Resim 8: cat acsess.log | cut -d " " -f 1,4,5,6,7,8,9,10,11,12,13,14,15,16,17 | grep "404"

cat acsess.log | cut -d " " -f 1,4,5,6,7,8,9,10,11,12,13,14,15,16,17 | grep "404"

Bu komut, acsess.log dosyasındaki her satırda belirli sütunları (1, 4-17) seçer ve ardından 404 hata kodlarını içeren satırları filtreler.

404 Hata Kodu Açıklaması:

Log kayıtlarında karşılaşılan 404 hataları, eksik veya yanlış yönlendirilmiş bağlantıları işaret eder. HTTP 404 Hata Kodu, web sunucusunun istemcinin talep ettiği URL'ye karşılık gelen sayfayı veya dosyayı bulamadığını belirtir. Bu hata, kullanıcı tarafından istenen kaynağın sunucuda mevcut olmadığını gösterir.

404 Hatalarının Sık Karşılaşılan Nedenleri:

- Kullanıcı tarafından girilen URL'nin yanlış veya eksik olması.
- Daha önce mevcut olan bir sayfanın silinmiş veya taşınmış olması.
- Web sitesindeki hatalı veya kırık bağlantılar.

Kırık Bağlantı (Broken Link): Bir web sayfasında yer alan ve tıklanabilir durumda olan ancak hedeflenen içeriğe veya sayfaya erişim sağlayamayan bir bağlantıdır. Kullanıcı bir kırık bağlantıya tıkladığında, istenen sayfa bulunamaz ve genellikle bir hata mesajı görüntülenir.

Analiz: Log dosyasında 404 hata kodlarının tespiti gerçekleştirilmiştir. grep "404" komutu, hata kodlarını içeren satırları filtreleyerek eksik veya yanlış yönlendirilmiş bağlantılarla ilgili sorunları belirlemeye yardımcı olmuştur. 404 hata kodlarının sıklığı ve detayları, eksik kaynaklar veya kırık bağlantılar hakkında bilgi sağlayarak, olası web sitesi sorunlarının analizi için temel oluşturmuştur. Bu tür hataların detaylı incelenmesi, web sitesinin kullanıcı deneyimini ve erişilebilirliğini iyileştirmek için önemlidir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log | cut -d " " -f 1,9 | grep "404" | sort | uniq -c | sort -nr
    7 61.14.246.6 404
    3 66.48.84.188 404
    2 36.40.116.182 404
    2 146.241.73.240 404
    1 193.230.127.66 404
```

Resim 9: `cat access.log | cut -d " " -f 1,9 | grep "404" | sort | uniq -c | sort -nr`

cat access.log | cut -d " " -f 1,9 | grep "404" | sort | uniq -c | sort -nr

Bu komut, access.log dosyasındaki IP adreslerini ve 404 hata kodlarını filtreler, her IP adresinin 404 hata sayısını hesaplar ve sonuçları en çok 404 hatası yapan IP adresinden başlayarak azalan sırada listeler.

Yapılan analiz sonucunda, aşağıdaki IP adresleri yüksek sayıda 404 hatasına neden olmuştur:

- **61.14.246[.]6:** 7 kez
- **66.48.84[.]188:** 3 kez
- **36.40.116[.]182:** 2 kez
- **146.241.73[.]240:** 2 kez
- **193.230.127[.]66:** 1 kez

Analiz: Log dosyasındaki 404 hata kodları incelemek, var olmayan sayfalara erişim denemeleri yapan ve bu hatalara neden olan IP adresleri tespit edilmiştir. Bu IP adreslerinin, özellikle **61.14.246[.]6** IP adresinin yüksek sayıda 404 hatası oluşturduğu görülmüştür. Bu durum, sistemdeki zayıf noktaları tespit etmeye yönelik tarama veya keşif faaliyetlerini işaret ediyor olabilir. Dolayısıyla, bu IP adreslerinin izlenmesi ve gerekirse engellenmesi önerilmektedir.

```
193.230.127.66 - - [25/Apr/2023:23:24:30 +0000] "POST /post/comment HTTP/1.1" 302 0 "http://victim.com/post?postId=3" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:24:31 +0000] "POST /post/comment HTTP/1.1" 302 0 "http://victim.com/post?postId=3" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:24:31 +0000] "POST /post/comment HTTP/1.1" 302 0 "http://victim.com/post?postId=3" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:24:50 +0000] "POST /post/comment HTTP/1.1" 302 0 "http://victim.com/post?postId=3" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:24:51 +0000] "GET /post/comment/confirmation?postId=3 HTTP/1.1" 200 880 "http://victim.com/post?postId=3" "Mozilla/5.0 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:27:51 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:27:53 +0000] "GET /my-account HTTP/1.1" 302 0 "http://victim.com/post/comment/confirmation?postId=3" "Mozilla/5.0 Safari/537.36"
193.230.127.66 - - [25/Apr/2023:23:27:54 +0000] "GET /login HTTP/1.1" 200 919 "http://victim.com/post/comment/confirmation?postId=3" "Mozilla/5.0 Safari/537.36"
146.241.73.240 - - [25/Apr/2023:23:27:54 +0000] "GET /resources/css/Xs.css HTTP/1.1" 200 4723 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [25/Apr/2023:23:29:54 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:40:10 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:40:11 +0000] "GET /footer HTTP/1.1" 404 31 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:40:14 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:42:36 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:42:37 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:42:37 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
146.241.73.240 - - [26/Apr/2023:21:42:37 +0000] "POST /login HTTP/1.1" 200 947 "http://victim.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4385.75 Safari/537.36"
```

Resim 20: cat acsess.log

cat acsess.log komutu ile log dosyasını incelerken, bazı aktiviteler dikkat çekmiş ve bu aktivitelerin analizi gerçekleştirilmiştir.

My-Account Erişimi

Log kaydında **193.230.127[.]66** IP adresine ait aktiviteler dikkat çekici bulunmuştur. Özellikle, **25 Nisan 2023, 23:27:53** tarihinde bu IP'den **GET /my-account** isteği gönderilmiş ve bu isteğe **HTTP 302** (Yönlendirme) yanıtı alınmıştır. Bu durum, kullanıcının "my-account" sayfasına erişim girişiminde bulunduğu ancak yönlendirildiğini göstermektedir.

Hemen ardından, **HTTP 200** (Başarılı) yanıtı dönen **POST /login** isteği gelmiştir. Bu da kullanıcının başarılı bir şekilde giriş yaptığı göstermektedir. Aynı zamanda, **Xs.css** dosyasının da bu süreçte başarılı bir şekilde yüklenmesiştir. Bu iki olayın arka arkaya gerçekleşmesi, saldırganın yetkisiz erişim girişiminde bulunabileceğini düşündürmektedir.

Analiz: Bu bulgular, yetkisiz erişim girişimlerinin ve potansiyel güvenlik açıklarının göstergesi olabilir. Özellikle **GET /my-account** isteği ve hemen ardından yapılan **POST /login** isteği, olası bir saldırganın "my-account" sayfasını hedef aldığı ve başarılı bir şekilde giriş yapmayı başardığını göstermektedir. Bu tür aktivitelerin dikkatle izlenmesi ve güvenlik önlemlerinin artırılması tavsiye edilmektedir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log | cut -d " " -f 9 | sort | uniq -c | sort -nr
204 200
187 302
15 404
```

Resim 11: cat acsess.log | cut -d “ “ -f 9 | sort | uniq -c | sort -nr

cat acsess.log | cut -d “ “ -f 9 | sort | uniq -c | sort -nr

Bu komut, `acsess.log` dosyasındaki HTTP yanıt kodlarını çıkarır, sayar ve en sık tekrar edenleri azalan sırada gösterir.

Log dosyasında tespit edilen HTTP durum kodlarının sayısı ve dağılımı şu şekildedir:

- **200 (Başarılı):** 204 kez
- **302 (Yönlendirme):** 187 kez
- **404 (Bulunamadı):** 15 kez

Analiz: Bu analiz sonucunda, log dosyasındaki HTTP durum kodlarının dağılımı belirlenmiştir. Analiz sonuçlarına göre, 200 (Başarılı) yanıt kodlarının yüksek sayıda oluşması, web sunucusunun genellikle başarılı yanıtlar verdiği ve kullanıcıların istenen sayfalara erişimde sorun yaşamadığını göstermektedir. 302 (Yönlendirme) kodlarının da oldukça sık görülmesi, kullanıcıların belirli sayfalara yönlendirildiğini ve yönlendirme işlemlerinin web sitesi için yaygın olduğunu ortaya koymaktadır. 404 (Bulunamadı) hata kodlarının sayısının görece düşük olması, mevcut sayfalara erişimde büyük bir sorun olmadığını işaret etmektedir. Bu, kullanıcıların eksik veya yanlış yönlendirilmiş sayfalarla sıkça karşılaşmadığını gösterir. Ancak, düşük sayıda 404 hatası bile dikkatli bir şekilde izlenmeli ve zaman zaman gözden geçirilmelidir. Çünkü herhangi bir eksik sayfa potansiyel kullanıcı deneyimi sorunlarına yol açabilir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr
372 "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
34 "Mozilla/5.0 (Macintosh; Intel Mac OS X
```

Resim 13: cat acsess.log | cut -d “ “ -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr

cat acsess.log | cut -d “ “ -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr

Bu komut, `acsess.log` dosyasındaki belirli sütunlardan (12-17) kullanıcı ajanı bilgilerini çıkarır, sıralar, her benzersiz kullanıcı ajanını sayar ve en sık tekrar edenleri azalan sırada listeler.

Analiz: Log dosyasında en sık görülen kullanıcı aracı, Mozilla/5.0 (Windows NT 10.0; Win64; x64) olmuştur.

“acsess.log” Dosyası Üzerinde Şüpheli Aktivitelerin Genel Değerlendirilmesi

Log kayıtları üzerinde yapılan incelemeler sonucunda, sunucuya yönelik potansiyel saldırılardan şüpheli aktiviteler tespit edilmiştir.

Tespit Edilen Şüpheli Aktiviteler:

- 404 Hataları:** Sık tekrarlanan 404 hataları, saldırganların var olmayan dosyaları arayarak sunucuyu taradığını gösterebilir.
- 302 Yönlendirmeleri:** Artan yönlendirmeler, yetkisiz sayfalara erişim denemeleri olabilir.
- Login İstekleri:** Tekrarlanan giriş denemeleri, brute-force saldırısı ihtimalini artırmaktadır.
- Hesap Yönetimi Sayfalarına Erişim:** "My-account" gibi sayfalara yapılan yetkisiz erişim denemeleri risklidir.

Öneriler:

- 404 Hataları:** Yüksek sayıda 404 hatası üreten IP adreslerinin aktiviteleri yakından izlenmeli ve gerektiğinde bu IP adresleri engellenmelidir.
- Brute-Force Saldırıları:** Brute-force saldırılara karşı ek güvenlik önlemleri (örneğin, captcha uygulamaları, IP engelleme) devreye alınmalıdır.
- Kritik Sayfalara Erişim:** "My-account" gibi kritik sayfalara yönelik erişim girişimlerinin detaylı olarak incelenmesi ve bu sayfalara yönelik güvenlik kontrollerinin sıklaştırılması gerekmektedir.

3) acsess.log.1 Analizi

Bu log dosyası, bir web sunucusuna ait erişim kayıtlarını detaylı olarak sunmaktadır. Her bir kayıt, istemcinin IP adresini, zaman damgasını, yapılan HTTP isteği (metod ve yol), sunucunun yanıt durumu kodunu, yanıt boyutunu, yönlendiren URL'yi ve kullanıcı aracını içermektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.1 | wc -l
307
```

Resim 14: cat acsess.log.1 | wc -l

cat acsess.log.1 | wc -l komutu kullanılarak, dosyanın toplamda 307 satır içerdiği tespit edilmiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.1 | grep "ERROR"
```

Resim 16: cat acsess.log.1 | grep "ERROR"

Log dosyasındaki hata mesajlarını analiz etmek amacıyla **grep "ERROR"** komutu kullanılmıştır. Ancak, bu komutun çalıştırılması sonucunda herhangi bir hata mesajı tespit edilmemiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.1 | cut -d " " -f 1 | sort | uniq -c | sort -r | head -n 5
56 119.221.17.90
39 234.161.112.162
28 48.124.217.56
24 21.94.54.79
21 69.90.24.5
```

Resim 17: cat acsess.log.1 | cut -d " " -f 1 | sort | uniq -c | sort -r | head -n 5

cat acsess.log.1 | cut -d " " -f 1 | sort | uniq -c | sort -r | head -n 5

Bu komut, acsess.log.1 dosyasındaki IP adreslerini çıkarır, sıralar, her bir IP'nin tekrar sayısını hesaplar ve en sık görülen ilk 5 IP adresini gösterir.

Bu sonuca göre, 119.221.17[.]90 IP adresi 56 kez tekrar ederek en sık karşılaşılan IP adresi olmuştur.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.1 | cut -d " " -f 1,7,9,10 | grep -i "login"
234.161.112.162 /login 200 978
234.161.112.162 /login?uid=test&pw=test 200 996
234.161.112.162 /login 200 1056
234.161.112.162 /login?uid=test&pw=test 200 1295
69.90.24.5 /wp-login 200 869
21.94.54.79 /acct_login 200 870
21.94.54.79 /customer_login 200 873
94.195.181.106 /smblogin 200 868
177.98.110.246 /snippets.gtl?uid=auto-login%2f 200 1000
```

Resim 18: cat access.log.1 | cut -d " " -f 1,7,9,10 | grep -i "login"

cut -d " " -f 1,7,9,10 | grep -i "login" komutu kullanılarak, cut komutu ile her satırda ilk, yedinci, dokuzuncu ve onuncu alanlar seçilmiş ve grep komutu ile bu alanlarda "login" terimi aramıştır. Arama, büyük/küçük harf duyarlılığı olmadan gerçekleştirılmıştır.

Log dosyasında yapılan inceleme sonucunda, çeşitli IP adreslerinden gelen şüpheli aktiviteler tespit edilmiştir. Analiz edilen aktiviteler şunları kapsamaktadır:

1. Aynı IP Adresinden Tekrarlayan Giriş Denemeleri:

Aynı IP adresinden tekrar eden giriş denemeleri yapılmıştır. Özellikle, 234.161.112[.]162 IP adresi, /login sayfasına birkaç kez erişim sağlamış ve bu istekler 200 yanıt kodu ile başarılı bir şekilde sonuçlanmıştır. Farklı parametrelerle yapılan bu giriş denemeleri, başarılı girişlerin yapılmış olabileceğini gösterse de, bu IP adresinin yüksek tekrar sıklığı şüpheli bir aktivite olarak değerlendirilebilir.

2. Farklı Giriş Sayfalarına Yapılan Erişimler:

Farklı giriş sayfalarına yapılan erişimler aşağıdaki gibi tespit edilmiştir:

- 69.90.24[.]5 IP adresi, /wp-login sayfasına erişim sağlamış ve yanıt kodu olarak 200 almıştır. Bu, WordPress tabanlı bir siteye yönelik olası bir brute-force saldırısını işaret edebilir.

- 21.94.54[.]79 IP adresi, /acct_login ve /customer_login gibi çeşitli login sayfalarına erişim sağlamıştır. Yanıt kodları 200'dür ve bu durum, IP adresinin farklı hesap giriş sayfalarını denediğini gösterebilir.
- 94.195.181[.]106 IP adresi, /smbLogin sayfasına erişim sağlamış ve yanıt kodu 200 olarak belirlenmiştir. Bu, SMB protokolü üzerinden yapılan olası bir brute-force girişimini işaret edebilir.

3. Otomatik Giriş Denemeleri:

Otomatik giriş denemeleri arasında şu durumlar gözlemlenmiştir:

177.98.110[.]246 IP adresi, /snippets.gtl?uid=autologin sayfasına erişim sağlamış ve yanıt kodu 200 olarak belirlenmiştir. Bu durum, "autologin" parametresi kullanılarak yapılan otomatik bir giriş denemesi olduğunu göstermektedir. Bu tür bir erişim denemesi, yetkisiz giriş girişimleri için bir işaret olabilir.

Analiz: Tekrarlayan giriş denemeleri ve çeşitli login sayfalarına yapılan erişimlerin sıklığı, bu IP adreslerinin şüpheli aktivitelerde bulunduğu gösterебilir. Bu aktiviteler, sisteminizde brute-force saldırısı veya diğer kötü niyetli girişimlerin olabileceği dair bir işaret olabilir. İlgili IP adreslerinin engellenmesi veya izlenmesi tavsiye edilir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.1 | grep "404"
```

Resim 19: cat acsess.log.1 | grep "404"

grep “404” komutu ile yapılan incelemede, log dosyasında herhangi bir 404 hata koduna rastlanmamıştır.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.1 | cut -d " " -f 9 | sort | uniq -c | sort -nr
303 200
3 302
1 413
1 304
```

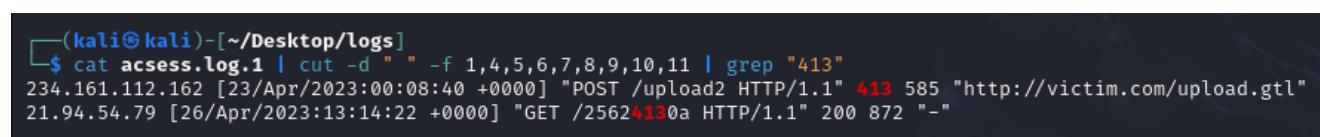
Resim 20: cat acsess.log | cut -d " " -f 9 | sort | uniq -c | sort -nr

`cat acesess.log | cut -d " " -f 9 | sort | uniq -c | sort -nr` komutu kullanılarak, HTTP durum kodlarının kaç kez tekrarlandığı analiz edilmiştir. Analiz sonucunda, HTTP durum kodlarının tekrar sayıları ve anlamları şu şekildedir:

- **200 OK:** 303 kez. İstek başarıyla işlenmiş ve yanıt doğru bir şekilde sağlanmıştır.
- **302 Found:** 3 kez. İstek yapılan kaynak geçici olarak başka bir adrese yönlendirilmiştir.
- **304 Not Modified:** 1 kez. Önceki yanıtla aynı içerik mevcut, bu nedenle yanıt gönderilmemiştir.
- **413 Payload Too Large:** 1 kez. Gönderilen veri boyutu sunucu tarafından işlenemeyecek kadar büyütür.

Analiz: HTTP durum kodlarının analizi, genellikle sorunsuz bir web sunucusu işleyişi göstermektedir. Özellikle **200 OK** kodunun yüksek sayıda olması, sunucunun çoğu isteği başarıyla işlediğini göstermektedir. **302 Found** kodu, geçici yönlendirmelerin olduğunu ve genellikle bu durumun önemli olmadığını işaret etmektedir. **304 Not Modified** ve **413 Payload Too Large** kodlarının düşük sikliği, sunucunun içerik güncellemeleri ve veri işleme kapasitesinin iyi bir seviyede olduğunu gösterir.

Ancak, **413 Payload Too Large** durum kodunun gözlemlenmiş olması, bazı kullanıcıların gönderdiği veri boyutlarının sunucunun işleyebileceğii sınırları aştığını belirtir. Bu durum, veri yükleme limitlerinin gözden geçirilmesi ve gerekli sınır ayarlarının yapılması gerektiğini işaret edebilir. Genel olarak, analiz, sunucunun genel performansının ve veri işleme kapasitesinin yeterli olduğunu, ancak belirli sınırların dikkatle yönetilmesi gerektiğini göstermektedir.



```
(kali㉿kali)-[~/Desktop/logs]
$ cat acesess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep "413"
234.161.112.162 [23/Apr/2023:00:08:40 +0000] "POST /upload2 HTTP/1.1" 413 585 "http://victim.com/upload.gtl"
21.94.54.79 [26/Apr/2023:13:14:22 +0000] "GET /25624130a HTTP/1.1" 200 872 "-"
```

Resim 21: `cat acesess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep '413'`

HTTP 413 durum kodunun geçtiği yerleri tespit etmek amacıyla `acesess.log.1` dosyasında yapılan incelemede, `cat acesess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep '413'` komutu kullanılmıştır. Bu komut, log dosyasındaki belirli alanları seçerek HTTP 413 durum kodunu içeren kayıtları belirler.

Analiz: Bu log kaydında, 234.161.112.162 IP adresinden yapılan POST isteği, /upload2 yoluna büyük bir veri yüklemeye yönelik bir işlem gerçekleştirmiştir. İstek, 413 Payload Too Large durum kodu ile yanıtlanmış olup, bu da gönderilen verinin sunucu tarafından kabul edilemeyecek kadar büyük olduğunu göstermektedir. Yanıt, 585 byte büyüklüğündedir ve istek <http://victim.com/upload.gtl> referans URL'si ile yapılmıştır. Bu durum, büyük dosya yüklemeleri ile ilgili bir sorunu veya potansiyel bir güvenlik tehdidini işaret ediyor olabilir.

Potansiyel Güvenlik Sorunları:

1. Denial of Service (DoS): Büyük dosya yüklemeleri, sunucunun kaynaklarını tüketebilir ve hizmetin yavaşlamasına veya kesilmesine neden olabilir.
2. Sistem Kaynaklarının Aşırı Kullanımı: Disk alanı ve CPU gibi kaynakların aşırı kullanımına yol açabilir, sunucu performansını etkileyebilir.
3. Kötü Amaçlı Yazılım: Yüklenen büyük dosyalar kötü amaçlı yazılımlar içerebilir ve sunucuya zarar verebilir.
4. Hedefe Yönelik Testler: Saldırganlar, sunucunun güvenlik açıklarını test etmek amacıyla büyük dosyalar yükleyebilir.

Bu sorunları ele almak için dosya yükleme limitlerini gözden geçirmek ve güvenlik önlemlerini artırmak önemlidir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep "234.161.112.162"
234.161.112.162 [23/Apr/2023:00:07:18 +0000] "GET / HTTP/1.1" 200 1226 "-"
234.161.112.162 [23/Apr/2023:00:07:26 +0000] "GET /login HTTP/1.1" 200 978 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:07:32 +0000] "GET /login?uid=test&pw=test HTTP/1.1" 200 996 "http://victim.com/login"
234.161.112.162 [23/Apr/2023:00:07:44 +0000] "GET /saveprofile?action=newguid=test&pw=test&is_author=True HTTP/1.1" 200 857 "http://victim.com/newaccount.gtl"
234.161.112.162 [23/Apr/2023:00:07:46 +0000] "GET /login HTTP/1.1" 200 1056 "http://victim.com/saveprofile?action=newguid=test&pw=test&is_author=True"
234.161.112.162 [23/Apr/2023:00:07:51 +0000] "GET /login?uid=test&pw=test HTTP/1.1" 200 1295 "http://victim.com/login"
234.161.112.162 [23/Apr/2023:00:07:52 +0000] "GET /snippets.gtl?uid=cheddar HTTP/1.1" 200 1294 "http://victim.com/login?uid=test&pw=test"
234.161.112.162 [23/Apr/2023:00:08:14 +0000] "GET /newsnippet?snipper=test+123 HTTP/1.1" 302 182 "http://victim.com/newsnippet.gtl"
234.161.112.162 [23/Apr/2023:00:08:14 +0000] "GET /snippets.gtl HTTP/1.1" 200 1227 "http://victim.com/newsnippet.gtl"
234.161.112.162 [23/Apr/2023:00:08:26 +0000] "GET /newsnippet.gtl HTTP/1.1" 200 1134 "http://victim.com/snippets.gtl"
234.161.112.162 [23/Apr/2023:00:08:33 +0000] "GET /newsnippet?snipper=test+23423423 HTTP/1.1" 302 182 "http://victim.com/newsnippet.gtl"
234.161.112.162 [23/Apr/2023:00:08:33 +0000] "GET /snippets.gtl HTTP/1.1" 200 1248 "http://victim.com/newsnippet.gtl"
234.161.112.162 [23/Apr/2023:00:08:35 +0000] "GET /upload.gtl HTTP/1.1" 200 1082 "http://victim.com/snippets.gtl"
234.161.112.162 [23/Apr/2023:00:08:40 +0000] "POST /upload2 HTTP/1.1" 413 585 "http://victim.com/upload.gtl"
234.161.112.162 [23/Apr/2023:00:08:40 +0000] "GET /favicon.ico HTTP/1.1" 304 0 "http://victim.com/upload2"
234.161.112.162 [23/Apr/2023:00:08:47 +0000] "GET / HTTP/1.1" 200 1326 "http://victim.com/upload.gtl"
234.161.112.162 [23/Apr/2023:00:08:48 +0000] "GET /snippets.gtl HTTP/1.1" 200 1248 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:08:50 +0000] "GET /newsnippet.gtl HTTP/1.1" 200 1134 "http://victim.com/snippets.gtl"
234.161.112.162 [23/Apr/2023:00:08:55 +0000] "GET /editprofile.gtl HTTP/1.1" 200 1362 "http://victim.com/newsnippet.gtl"
234.161.112.162 [23/Apr/2023:00:08:59 +0000] "GET / HTTP/1.1" 200 1326 "http://victim.com/editprofile.gtl"
234.161.112.162 [23/Apr/2023:00:09:02 +0000] "GET /snippets.gtl?uid=test HTTP/1.1" 200 1208 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:09:09 +0000] "GET /snippets.gtl?uid=brie HTTP/1.1" 200 1252 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:09:16 +0000] "GET / HTTP/1.1" 200 1324 "http://victim.com/snippets.gtl?uid=brie"
234.161.112.162 [23/Apr/2023:00:10:01 +0000] "GET / HTTP/1.1" 200 1255 "-"
234.161.112.162 [23/Apr/2023:00:10:01 +0000] "GET /lib.js HTTP/1.1" 200 805 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:10:02 +0000] "GET /cheese.png HTTP/1.1" 200 9993 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:10:02 +0000] "GET /favicon.ico HTTP/1.1" 200 430 "http://victim.com/"
234.161.112.162 [23/Apr/2023:00:10:12 +0000] "GET / HTTP/1.1" 200 1253 "-"
234.161.112.162 [25/Apr/2023:17:21:01 +0000] "GET /test HTTP/1.1" 200 866 "-"
234.161.112.162 [25/Apr/2023:17:21:01 +0000] "GET /ohacker94526032f HTTP/1.1" 200 875 "-"
234.161.112.162 [25/Apr/2023:17:21:01 +0000] "GET /ebs%2f HTTP/1.1" 200 867 "-"
234.161.112.162 [25/Apr/2023:17:21:01 +0000] "GET /PANG123456789%2f HTTP/1.1" 200 876 "-"
234.161.112.162 [25/Apr/2023:17:21:02 +0000] "GET /autodesk%2f HTTP/1.1" 200 870 "-"
234.161.112.162 [25/Apr/2023:17:21:02 +0000] "GET /csb%2f HTTP/1.1" 200 867 "-"
234.161.112.162 [25/Apr/2023:17:21:02 +0000] "GET /kumon%2f HTTP/1.1" 200 869 "-"
234.161.112.162 [25/Apr/2023:17:21:02 +0000] "GET /supervise%2f HTTP/1.1" 200 871 "-"
234.161.112.162 [25/Apr/2023:17:21:02 +0000] "GET /sec%2f HTTP/1.1" 200 867 "-"


```

Resim 22: cat acsess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep "234.161.112[.]162"

Bu IP adresini daha detaylı incelemek amacıyla **cat acsess.log.1 | cut -d " " -f 1,4,5,6,7,8,9,10,11 | grep "234.161.112[.]162"** komutu kullanılarak log kayıtları taramıştır.

Analiz: Elde edilen veriler doğrultusunda 234.161.112[.]162 IP adresi 23 Nisan 2023'te çeşitli web sayfalarına normal kullanıcı etkileşimleri gibi görünen isteklerde bulunmuştur. Ancak, 25 Nisan 2023'te aynı IP adresi, şüpheli ve rastgele görünümlü URL'lere birçok GET isteği yapmıştır. Bu durum, potansiyel bir güvenlik taraması veya denemesi olabilir.

Potansiyel Güvenlik Sorunları:

- Şüpheli URL Erişimleri:** Rastgele görünen URL'ler, güvenlik açılarını test etmeye yönelik olabilir.
- Olası Güvenlik Taramaları:** URL'lerin olağanlığı olması, potansiyel tarama veya kaynak tüketimi girişimleri anlamına gelebilir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.1 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10
 11 /
 10 /snippets.gtl?uid=test
  6 /newsnippet.gtl
  5 /snippets.gtl
  3 /upload.gtl
  2 /test
  2 /snippets.gtl?uid=%2e%2e%2f
  2 /login?uid=test&pw=test
  2 /login
  2 /favicon.ico
```

Resim 23: cat acsess.log.1 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10

cat acsess.log.1 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10

Komutu kullanılarak yapılan analizde, sunucunun kök dizinine (/) 11 kez erişildiği tespit edilmiştir.

Analiz: Kök dizine yapılan erişimler, genellikle web sitesinin ana sayfasına ziyaretleri gösterir ve çoğu zaman normal bir trafik göstergesi olarak kabul edilir. Bu erişimler, kullanıcıların ana sayfayı ziyaret ettiği veya otomatik sistemlerin tarama yaptığı durumları yansıtabilir. Ancak, bu tür erişimlerin sayısında ani bir artış gözlemlenirse bu durum, sunucuya yönelik potansiyel bir keşif veya kötü niyetli tarama aktivitesini işaret edebilir. Bu nedenle, kök dizine yapılan erişimlerin sıklığı ve zamanlaması dikkatle izlenmelidir.

Ayrıca, /snippets.gtl?uid=test URL'sine 10 kez erişim sağlandığı belirlenmiştir. URL'deki uid=test ifadesi, olası bir brute-force veya deneme yanlışma saldırısını işaret edebilir. Brute-force saldırıları genellikle çeşitli uid ve pw kombinasyonları kullanarak yetkisiz erişim elde etmeye yönelik denemeler içerir. Bu tür aktivitelerin kısa bir süre içinde yoğun olarak yapılması, potansiyel bir saldırıyı gösterebilir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.1 | cut -d " " -f 1,4,7 | grep "/snippets.gtl?uid=test"
234.161.112.162 [23/Apr/2023:00:09:02 /snippets.gtl?uid=test
21.94.54.79 [26/Apr/2023:13:14:33 /snippets.gtl?uid=test
21.94.54.79 [26/Apr/2023:13:14:36 /snippets.gtl?uid=test
204.218.128.123 [23/Apr/2023:00:15:35 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:15:43 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:15:51 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:16:14 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:16:57 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:16:57 /snippets.gtl?uid=test
123.114.236.235 [23/Apr/2023:00:18:29 /snippets.gtl?uid=test
156.139.188.182 [23/Apr/2023:00:18:48 /snippets.gtl?uid=testing%2f
```

Resim 24: `cat acsess.log.1 | cut -d " " -f 1,4,7 | grep "/snippets.gtl?uid=test"`

En çok tekrar eden URL'lerden biri olan `/snippets.gtl?uid=test` adresini daha detaylı incelemek amacıyla, `cat acsess.log.1 | cut -d " " -f 1,4,7 | grep "/snippets.gtl?uid=test"` komutu kullanılmıştır. Bu komut, log dosyasında bu belirli URL'ye yapılan istekleri ve ilgili IP adreslerini içeren kayıtları filtreleyerek detaylı bir analiz sağlar.

Analiz: Yapılan analiz sonucunda, `/snippets.gtl?uid=test` URL'sine erişimlerin çeşitli IP adreslerinden yapıldığı tespit edilmiştir. Bu durum, URL'ye yönelik potansiyel bir brute-force saldırısı veya kötü niyetli erişim denemelerini işaret edebilir. Erişimlerin yoğunluğu ve farklı IP adreslerinden yapılması, bu URL'nin hedef alındığını ve potansiyel bir saldırı belirtisi olabileceğini gösterir. Bu tür aktivitelerin detaylı bir şekilde analiz edilmesi ve gerekli güvenlik önlemlerinin alınması, sistemin güvenliğini korumak için önemlidir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.1 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr
264 "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
44 "Mozilla/5.0 (Macintosh; Intel Mac OS X
```

Resim 25: `cat acsess.log.1 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr`

`cat acsess.log.1 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr`

Bu komut, `acsess.log.1` dosyasındaki belirli sütunlardan (12-17) kullanıcı ajanı bilgilerini çıkarır, sıralar, her benzersiz kullanıcı ajanını sayar ve en sık tekrar edenleri azalan sıradaki listeler.

Analiz: Log dosyasında en sık görülen kullanıcı aracı, Mozilla/5.0 (Windows NT 10.0; Win64; x64) olmuştur.

“acsess.log.1” Dosyası Üzerinde Şüpheli Aktivitelerin Genel Değerlendirilmesi

acsess.log.1 dosyasının analizi, sunucunuza yapılan HTTP isteklerinin detaylı bir incelemesini sağlamıştır. Bu log dosyası, sunucunuza gelen isteklerin IP adresleri, erişilen URL'ler ve alınan yanıt kodları hakkında bilgi sunmaktadır. Yapılan analiz, genel trafik hacmi ve kullanıcı davranışları hakkında değerli bilgiler sağlamıştır. Özellikle belirli URL'lere yapılan yoğun erişimler ve tekrarlayan giriş denemeleri dikkat çekmiştir.

Şüpheli Aktiviteler:

1. Tekrarlayan Giriş Denemeleri:

- IP adresi 234.161.112[.]162, `/login` sayfasına birkaç kez tekrarlayan isteklerde bulunmuştur.
- Bu istekler farklı parametrelerle yapılmış olup, olası bir brute-force saldırısı veya yetkisiz erişim denemesi olabilir.

2. Farklı Login Sayfalarına Yoğun Erişim:

- IP adresleri 21.94.54[.]79 ve 94.195.181[.]106, çeşitli login sayfalarına (`/acct_login`, `/customer_login`, `/smbLogin`) yoğun erişim sağlamıştır.
- Bu tür erişimler, sistemdeki farklı hesapları hedef alan bir saldırı girişimini gösterebilir.

3. Kök Dizin (/) ve Spesifik URL'lere Yoğun Erişim:

- Sunucunun kök dizinine (`/`) 11 kez ve `/snippets.gtl?uid=test` URL'sine 10 kez erişim yapılmıştır.
- Özellikle belirli parametrelerle yapılan bu yoğun erişimler, olası bir keşif veya saldırı girişiminin belirtisi olabilir.

Öneriler:

1. IP Tabanlı İzleme ve Engelleme:

- Şüpheli aktivitelerde bulunan IP adresleri izlenmeli ve gereklse geçici olarak engellenmelidir.
- Tekrarlayan başarısız giriş denemeleri gibi anormal davranışlar gösteren IP'ler kara listeye alınmalıdır.

2. Güvenlik Duvarı ve WAF (Web Application Firewall) Ayarları:

- Sunucunun kök dizinine ve belirli URL'lere yönelik yoğun istekler için güvenlik duvari kuralları gözden geçirilmelidir.
- Bu tür istekler için uyarı sistemleri yapılandırılmalıdır.
- WAF kullanarak, potansiyel saldırıları daha erken aşamada tespit etmek ve engellemek mümkün olabilir.

3. Brute-Force Koruması:

- Login sayfaları için brute-force koruması sağlamak amacıyla CAPTCHA gibi ek güvenlik önlemleri eklenmelidir.
- Belirli sayıda başarısız deneme sonrası geçici kilitleme mekanizması uygulanmalıdır.

4. Düzenli Log Analizi:

- Log dosyalarının düzenli olarak analiz edilmesi, şüpheli aktivitelerin erken tespiti için kritik öneme sahiptir.
- Otomatik uyarı sistemleri kurarak, belirli bir süre içinde anormal trafik artışı veya şüpheli aktiviteler tespit edildiğinde anında müdahale edilmelidir.

4) acsess.log.2 Analizi

Bu log dosyası, bir web sunucusuna yapılan erişimlerin ayrıntılarını sunmaktadır. Her kayıt, istemcinin IP adresini, zaman damgasını, gerçekleştirilen HTTP isteğini (metod ve yol), sunucunun yanıt durum kodunu, yanıt boyutunu, yönlendiren URL'yi ve kullanıcı aracını içermektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | wc -l
237
```

Resim 26: `cat acsess.log.1 | wc -l`

`cat acsess.log.1 | wc -l` komutu kullanılarak yapılan incelemede, `acsess.log.1` dosyasının toplamda 237 satır içerdiği tespit edilmiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | grep "ERROR"
```

Resim 27: `cat acsess.log.2 | grep "ERROR"`

`grep "ERROR"` komutu kullanılarak yapılan analiz sonucunda, belirtilen log dosyasında herhangi bir hata mesajı tespit edilememiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | cut -d " " -f 1 | sort | uniq -c | sort -nr
119 86.236.188.85
76 254.198.150.19
24 182.195.27.49
10 206.52.45.12
9 178.78.113.5
```

Resim 28: `cat acsess.log.1 | cut -d " " -f 1 | sort | uniq -c | sort -nr`

cat acsess.log.1 | cut -d " " -f 1 | sort | uniq -c | sort -nr komutu kullanılarak, acsess.log.2 dosyasındaki IP adreslerinin her birinin tekrar sayıları belirlenmiş ve en çok tekrar eden IP adresleri azalan sırada sıralanmıştır.

Analiz: Bu analiz sonucunda **86.236.188[.]85** IP adresinin **119** kez tekrar ettiği ve en çok karşılaşılan IP adresi olduğu belirlenmiştir.

Resim 29: `cat acsess.log.2 | cut -d " " -f 1,4,5,6,7,8,9,10 | grep "86.236.188[.]85"`

cat acsess.log.2 | cut -d " " -f 1,4,5,6,7,8,9,10 | grep “86.236.188[.]85” komutu, acsess.log.2 dosyasındaki her satırda belirli alanları seçip 86.236.188[.]85 IP adresine ait kayıtları filtreler. Bu işlem, ilgili IP adresinin sunucuda gerçekleştirdiği isteklerin detaylarını gösterir.

Analiz: En fazla tekrar eden IP adresi olarak belirlenen 86.236.188[.]85 adresinin, çoğunlukla HTTP durum kodu 400 dönen isteklerde yer aldığı gözlemlenmiştir. Bu durum, bu IP adresinden gelen isteklerin büyük kısmının, sunucu tarafından hatalı veya geçersiz olarak değerlendirilmiş olduğunu göstermektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log.2 | cut -d " " -f 1,7,9,10 | grep -i "login"
```

Resim 30: cat access.log.2 | cut -d " " -f 1,7,9,10 | grep -i "login"

Login terimini aramak amacıyla **cat access.log.2 | cut -d " " -f 1,7,9,10 | grep -i "login"** komutu kullanılmıştır. Ancak, bu arama sonucunda herhangi bir eşleşme bulunamamıştır.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log.2 | cut -d " " -f 9 | sort | uniq -c | sort -nr
 120 400
  59 200
  34 499
  23 404
    2 302
```

Resim 31: cut -d " " -f 9 | sort | uniq -c | sort -nr

cut -d " " -f 9 | sort | uniq -c | sort -nr komutu kullanılarak, her satırda dokuzuncu alan seçilmiş, ardından bu alanlardaki değerler sıralanmış, her bir değer için tekrar sayıları hesaplanmış ve sonuçlar azalan sırada sıralanmıştır.

Yapılan analizde, HTTP durum kodlarının sıklıkları aşağıdaki gibi tespit edilmiştir:

- **400 (Bad Request):** 120 kez. Bu, istemcilerden gelen hatalı isteklerin yüksek olduğunu gösterir.
- **200 (OK):** 59 kez. İsteklerin başarıyla işlendiğini ve doğru yanıt verildiğini belirtir.
- **499 (Client Closed Request):** 34 kez. İstemcilerin bağlantıyı kapattığını ve genellikle zaman aşımından kaynaklandığını işaret eder.
- **404 (Not Found):** 23 kez. İstenilen kaynağın bulunamadığını gösterir.
- **302 (Found):** 2 kez. Geçici yönlendirmeleri belirtir.

Analiz: Genel olarak, 400 kodlu hataların yüksek oranı dikkat çekicidir ve kullanıcıların doğru istekleri göndermesini sağlamak için ek kontroller veya iyileştirmeler yapılması gerektiğini gösterir. Diğer durum kodları, sunucunun genel performansını ve kullanıcı etkileşimlerini yansıtır. Bu tür analizlerin düzenli olarak yapılması, sistem performansını artırmak ve kullanıcı deneyimini iyileştirmek için önemli bir adımdır.

Resim 32: `cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9 | grep "400"`

HTTP durum kodu 400 olan istekler, **grep “400”** komutu ile filtrelenmiştir. Bu yöntem, log dosyasındaki 400 hata kodunu içeren kayıtları seçmek ve incelemek amacıyla kullanılmıştır.

Analiz: Analiz sonucunda, bu isteklerin URL kodlaması kullanılarak yapıldığı tespit edilmiştir. URL kodlaması, özel karakterlerin ve boşlukların güvenli ve uyumlu bir şekilde temsil edilmesini sağlayan bir yöntemdir. Bu işlemde, özel karakterler ve boşluklar, % işaretü ile başlayan ve ardından iki hexadecimal rakam içeren bir biçimde kodlanır.

Bu tür URL kodlaması, genellikle URL içinde özel karakterlerin güvenli bir şekilde temsil edilmesi için kullanılır. Ancak, kötü niyetli girişler veya saldırılar için de kullanılabilir. URL kodlaması ile temsil edilen karakterlerin dikkatli bir şekilde analizi, potansiyel güvenlik açıklarını veya anormal girişleri belirlemek için önemlidir.

```

/post?postId=%2f%2f%2f%2fexample%2ecom%2f
/post?postId=%2f%2f%2f%2fexample%2ecom%2f%2e%2e
/post?postId=%2f%2f%2f%2fexample%2ecom%2f%2e%2e%2f
/post?postId=%2f%2f%2f%2fexample%2ecom%2f%2f%2e%2e
/post?postId=9
/post?postId=%2f%2f%2f%2f%2fexample%2ecom%2f%2f
/post?postId=%2f%2f%2f%5c%3b@example%2ecom
/post?postId=%2f%2f%2f%2fgoogle%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2f%2fexample%2ecom
/post?postId=%2f%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2f%2fexample%2ecom%2f
/post?postId=%2f%2f%2f%2fgoogle%2ecom%2f%2f%2e%2e%2f encoded JavaScript URLs from complete
/post?postId=%2f%2f%2f%2f%2fgoogle%2ecom%2f%2f%2e%2e%2f URL Decoder/Encoder for offline us
/post?postId=%2f%2f%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e Attribution-ShareAlike
/post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=%2fhttps%3a%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=%2f%2fhttps%3a%2f%2fwww%2ewhitelisteddomain%2etld@google%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2fwww%2egoogle%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2fwww%2egoogle%2ecom%2f%2f%2e%2e%2f www%2egoogle%2ecom%2f%2f%2e%2e
/post?postId=%2f%2f%2fwww%2egoogle%2ecom%2f%2f%2e%2e
/post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2f%2e%2e

```

Resim 33: cat acsess.log.2 | cut -d " " -f 7

cat acsess.log.2 | cut -d " " -f 7 komutu kullanılarak, log dosyasının 7. sütunundaki URL adresleri çıkarılmıştır. Bu URL'ler daha sonra decode edilmiştir.

```

/post?postId=///example.com///
/post?postId=///\;@example.com
/post?postId=///google.com//..
/post?postId=///example.com
/post?postId=///www.whitelisteddomain.tld@google.com//..
/post?postId=///example.com/
/post?postId=///google.com//..
/post?postId=///google.com//..
/post?postId=///www.whitelisteddomain.tld@google.com//..
/post?postId=///www.whitelisteddomain.tld@google.com//..
/post?postId=https://google.com//..
/post?postId=https://www.whitelisteddomain.tld@google.com//..
/post?postId=https://google.com//..
/post?postId=//www.google.com//..
/post?postId=//www.whitelisteddomain.tld@www.google.com//..
/post?postId=//www.whitelisteddomain.tld@www.google.com//..
/post?postId=///www.google.com//..
/post?postId=///www.google.com//..
/post?postId=https://www.google.com//..

```

Resim 34: URL Decode

Decode işlemi sonucunda aşağıdaki URL'ler elde edilmiştir:

- **google.com**
- **example.com**
- **whitelisteddomain[.]tld@google.com**

Bu işlem, URL'lerin daha net bir biçimde analiz edilmesi ve içeriklerinin doğru bir şekilde değerlendirilmesi amacıyla yapılmıştır. Decode edilmiş URL'ler, güvenlik ve trafik analizleri için önemli bilgiler sunabilir.

Analiz: www.whitelisteddomain[.]tld@google.com şeklindeki bir URL'ye yapılan istekler, genellikle güvenlik testi veya konfigürasyon doğrulama amacı taşıyabilir. Ancak, bu tür bir URL'ye yapılan istekler aynı zamanda anormal ve potansiyel olarak zararlı girişimler de olabilir. Özellikle @ işaretti, eski kimlik doğrulama yöntemlerinde kullanıcı adı ve şifre bilgilerini taşıyabiliyordu ve modern güvenlik standartlarında bu tür yöntemler önerilmemektedir. Bu nedenle, bu tür URL'lere yapılan isteklerin dikkatle izlenmesi ve değerlendirilmesi önemlidir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9,10,11 | grep "404"
254.198.150.19 [26/Apr/2023:19:44:47 "GET /post?postId=0 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:09 "GET /post?postId=0 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:11 "GET /post?postId=16 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:12 "GET /post?postId=15 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:12 "GET /post?postId=14 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:12 "GET /post?postId=13 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:12 "GET /post?postId=12 HTTP/1.1" 404 31 "http://victim.com/"
254.198.150.19 [26/Apr/2023:19:45:12 "GET /post?postId=11 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:09 "GET /post?postId=0 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:09 "GET /post?postId=2010 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:09 "GET /post?postId=2011 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2020 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2019 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2018 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2017 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2015 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2016 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2014 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2013 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:11 "GET /post?postId=2012 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:13 "GET /post?postId=2023 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:13 "GET /post?postId=2022 HTTP/1.1" 404 31 "http://victim.com/"
182.195.27.49 [27/Apr/2023:19:46:13 "GET /post?postId=2021 HTTP/1.1" 404 31 "http://victim.com/"
```

Resim 35: cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9 | grep "404"

grep "404" komutu ile yapılan arama sonucunda, belirli bir kaynak yolu üzerinde sistematik olarak farklı postId değerleriyle yapılan GET istekleri ve her istekte alınan 404 Not Found yanıtları gözlemlenmiştir. Bu tür bir davranış aşağıdaki nedenlerle potansiyel bir saldırı göstergesi olabilir:

- **Brute Force Attack:** Belirli bir kaynak üzerinde sistematik olarak farklı postId değerleriyle yapılan denemeler, potansiyel bir brute force saldırısına işaret edebilir. Bu durumda, saldırgan mevcut olmayan kaynakları bulmaya veya sunucunun zayıf noktalarını keşfetmeye çalışıyor olabilir.
- **Keşif ve Bilgi Toplama:** Bu tür istekler, bir saldırganın sistem hakkında daha fazla bilgi toplama amacı taşıyor olabilir. 404 Not Found yanıtlarının sürekli alınması, saldırganın sistemin yapılandırması hakkında bilgi edinmeye çalıştığını ve sistemde mevcut olan veya olmayan kaynakları anlamaya yönelik girişimlerde bulunduğu gösterebilir.
- **Zayıflık Taraması:** Farklı postId değerleriyle yapılan bu denemeler, sunucunun yanıt verme biçimini anlamak ve potansiyel güvenlik açıklarını keşfetmek amacıyla yapılmış olabilir. Saldırgan, sistemin güvenlik açıklarını tespit etmek için farklı parametrelerle çeşitli testler gerçekleştiriyor olabilir.

Analiz: Bu tür sistematik ve tekrarlayan 404 Not Found yanıtları, sistemde potansiyel güvenlik açıklarını belirlemek veya sistem hakkında bilgi toplamak amacıyla gerçekleştirilen şüpheli aktivitelerin bir parçası olabilir. Dolayısıyla, bu tür davranışların dikkatle izlenmesi ve analiz edilmesi, potansiyel saldırıları tespit etmek ve güvenlik önlemlerini güçlendirmek açısından önemlidir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.2 | cut -d " " -f 1,4,5,6,7,8,9,10 | grep "200"
178.78.113.5 [18/Apr/2023:19:42:00 +0000] "GET / HTTP/1.1" 200 2270
178.78.113.5 [18/Apr/2023:19:42:04 +0000] "GET /post?postId=9 HTTP/1.1" 200 2965
178.78.113.5 [18/Apr/2023:19:42:05 +0000] "GET /post?postId=6 HTTP/1.1" 200 3013
178.78.113.5 [18/Apr/2023:19:42:06 +0000] "GET /post?postId=7 HTTP/1.1" 200 3189
178.78.113.5 [18/Apr/2023:19:42:07 +0000] "GET /post?postId=8 HTTP/1.1" 200 2675
178.78.113.5 [18/Apr/2023:19:42:08 +0000] "GET /post?postId=3 HTTP/1.1" 200 2940
178.78.113.5 [18/Apr/2023:19:44:09 +0000] "GET /post?postId=2 HTTP/1.1" 200 3203
178.78.113.5 [18/Apr/2023:19:48:10 +0000] "GET /post?postId=5 HTTP/1.1" 200 2662
178.78.113.5 [18/Apr/2023:39:42:12 +0000] "GET /post?postId=4 HTTP/1.1" 200 3179
206.52.45.12 [18/Apr/2023:19:42:13 +0000] "GET /post?postId=1 HTTP/1.1" 200 3093
206.52.45.12 [18/Apr/2023:49:42:17 +0000] "GET /post?postId=10 HTTP/1.1" 200 2542
206.52.45.12 [18/Apr/2023:19:42:42 +0000] "GET /post/comment/confirmation?postId=9 HTTP/1.1" 200 914
206.52.45.12 [18/Apr/2023:19:42:46 +0000] "GET /post?postId=9 HTTP/1.1" 200 2983
206.52.45.12 [18/Apr/2023:19:42:47 +0000] "GET / HTTP/1.1" 200 2270
206.52.45.12 [18/Apr/2023:19:42:52 +0000] "GET /post?postId=6 HTTP/1.1" 200 3013
206.52.45.12 [23/Apr/2023:19:43:12 +0000] "GET /post/comment/confirmation?postId=6 HTTP/1.1" 200 913
206.52.45.12 [23/Apr/2023:19:43:29 +0000] "GET /post?postId=8 HTTP/1.1" 200 2675
254.198.150.19 [26/Apr/2023:03:44:31 +0000] "GET /post?postId=3 HTTP/1.1" 200 2940
254.198.150.19 [26/Apr/2023:03:44:36 +0000] "GET /post?postId=2 HTTP/1.1" 200 3203
254.198.150.19 [26/Apr/2023:03:44:46 +0000] "GET /post?postId=4 HTTP/1.1" 200 3179
254.198.150.19 [26/Apr/2023:03:44:47 +0000] "GET /post?postId=10 HTTP/1.1" 200 2542
254.198.150.19 [26/Apr/2023:03:44:49 +0000] "GET /post?postId=10 HTTP/1.1" 200 2542
254.198.150.19 [26/Apr/2023:03:44:50 +0000] "GET / HTTP/1.1" 200 2270
254.198.150.19 [26/Apr/2023:03:44:52 +0000] "GET /post?postId=1 HTTP/1.1" 200 3093
254.198.150.19 [26/Apr/2023:03:44:52 +0000] "GET / HTTP/1.1" 200 2270
254.198.150.19 [26/Apr/2023:03:44:55 +0000] "GET / HTTP/1.1" 200 2270
254.198.150.19 [26/Apr/2023:03:44:56 +0000] "GET / HTTP/1.1" 200 2270
254.198.150.19 [26/Apr/2023:03:44:58 +0000] "GET /post?postId=10 HTTP/1.1" 200 2542
```

Resim 36: cat acsess.log.2 | cut -d " " -f 1,4,5,6,7,8,9 | grep "200"

grep “200” komutu kullanarak yapılan incelemede isteklerin 200 OK dönmesi, sunucunun bu istekleri başarıyla işlediğini ve belirli postId değerleri için talep edilen verilerin başarıyla sağlandığını gösterir. Bu durum, hem veri erişiminin düzgün şekilde sağlandığını hem de sunucunun istekleri doğru bir şekilde işlediğini ifade eder.

Analiz: Kısa bir süre içinde farklı postId değerlerine yapılan başarılı isteklerin hızlı bir şekilde gerçekleşmesi, şüpheli bir durum olarak değerlendirilebilir. Bu, otomatik saldırılar, test amaçlı işlemler veya potansiyel güvenlik açılarını araştırma gibi çeşitli nedenlerden kaynaklanabilir.



```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.2 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10
19 /
15 /post?postId=2
11 /post?postId=10
9 /post?postId=1
8 /post?postId=9
5 /post?postId=8
5 /post?postId=6
5 /post?postId=5
5 /post?postId=4
5 /post?postId=3
```

Resim 37: `cat access.log.2 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10`

En fazla erişim sağlanan URL'leri belirlemek amacıyla **cat access.log.2 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 10** komutu kullanılmıştır.

Analiz: Bu analiz sonucunda, sunucunun kök dizinine (/) toplam 19 kez erişim sağlandığı tespit edilmiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | cut -d " " -f 1,4,7,8,9,10 | grep "post?postId=2"
178.78.113.5 [18/Apr/2023:19:44:09 /post?postId=2 HTTP/1.1" 200 3203
254.198.150.19 [26/Apr/2023:03:44:33 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:33 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:33 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:34 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:35 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:35 /post?postId=2 HTTP/1.1" 499 0
254.198.150.19 [26/Apr/2023:03:44:36 /post?postId=2 HTTP/1.1" 200 3203
254.198.150.19 [26/Apr/2023:19:44:49 /post?postId=2 HTTP/1.1" 200 3203
254.198.150.19 [26/Apr/2023:19:45:11 /post?postId=2 HTTP/1.1" 200 3203
```

Resim 38: `cat acsess.log.2 | cut -d " " -f 1,4,7,8,9,10 | grep "post?postId=2"`

`cat acsess.log.2 | cut -d " " -f 1,4,7,8,9,10 | grep "post?postId=2"` komutu, acsess.log.2 dosyasındaki her satırda belirli alanları seçip `post?postId=2` sorgusunu içeren kayıtları filtreler. Bu işlem, bu URL ile yapılan isteklerin detaylarını gösterir.

Analiz: Belirli postId değerlerine kısa sürede yapılan yüksek hacimli ve hedefli erişimler, potansiyel bir saldırı veya kötü niyetli etkinliklerin işaretini olabilir. Bu tür aktivitelerin dikkatlice izlenmesi, analiz edilmesi ve gerekli güvenlik önlemlerinin alınması büyük önem taşır.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.2 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10
33 [27/Apr/2023:15:45:25
31 [27/Apr/2023:15:45:24
29 [27/Apr/2023:15:45:23
10 [27/Apr/2023:15:45:26
9 [27/Apr/2023:19:46:11
9 [27/Apr/2023:15:45:22
6 [27/Apr/2023:15:45:35
6 [26/Apr/2023:19:45:12
6 [26/Apr/2023:03:44:52
5 [26/Apr/2023:03:44:34
```

Resim 39: `cat acsess.log.2 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10`

En sık tekrar eden tarihler üzerine yapılan analizde, `cat acsess.log.2 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10` komutu kullanılarak en fazla tekrar eden 10 tarih belirlenmiştir.

Analiz: Bu analiz sonucunda, 27/Apr/2023:15:45:25 tarihinin toplam 33 kez tekrar ettiği tespit edilmiştir. Bu tarihi daha detaylı analiz etmek amacıyla ek bir inceleme yapılmıştır.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9 | sort | uniq -c | sort -nr | grep "27/Apr/2023:15:45:25"
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e%2f HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e%2f HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2ewhitelisteddomain%2etld@www%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
1 86.236.188.85 [27/Apr/2023:15:45:25 "GET /post?postId=https%3a%2f%2fwww%2egoogle%2ecom%2f%2e%2e HTTP/1.1" 400
```

Resim 40: cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9 | sort | uniq -c | sort -nr | grep "27/Apr/2023:15:45:25"

cat acsess.log.2 | cut -d " " -f 1,4,6,7,8,9 | sort | uniq -c | sort -nr | grep "27/Apr/2023:15:45:25"

Komutu, acsess.log.2 dosyasındaki belirli alanları (-f 1,4,6,7,8,9) seçer, verileri sıralar, tekrar sayılarına göre düzenler ve 27 Nisan 2023, 15:45:25 tarihli kayıtları filtreler. Bu işlem, o tarihteki IP adresi, URL ve yanıt kodları gibi bilgilerin dağılımını gösterir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.2 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr
238 "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
```

Resim 41: cat acsess.log.2 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr

```
cat acsess.log.2 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr
```

Bu komut, acsess.log.2 dosyasındaki 12 ila 17. sütunlardaki kullanıcı ajanı bilgilerini çıkarır, sıralar, tekrar sayılarını hesaplar ve en sık tekrar edenleri öne alarak sıralar.

Analiz: Analiz sonucunda, log dosyasında en sık görülen kullanıcı ajanı Mozilla/5.0 (Windows NT 10.0; Win64; x64) olarak belirlenmiştir. Bu, en çok kullanılan tarayıcı ve işletim sistemi kombinasyonunu temsil eder.

“acsess.log.2” Dosyası Üzerinde Şüpheli Aktivitelerin Genel Değerlendirilmesi

acsess.log.2 dosyası üzerinden yapılan analizler, sunucuya yönelik potansiyel şüpheli aktiviteleri ve anomal davranışları ortaya koymaktadır.

Başlıca Bulgular:

- IP Adresleri:** En çok istek yapan IP adresleri arasında **86.236.188[.]85**, **254.198.150[.]19** ve **182.195.27[.]49** yer almaktadır. Özellikle **86.236.188[.]85** IP adresinden toplamda 119 istek yapılmıştır.
- URL'ler:** En sık erişilen URL, ana sayfanız olan / olup, bu URL'ye 19 kez erişilmiştir. Ayrıca, **postId=2** ve **postId=10** gibi spesifik içeriklere yönelik talepler de yüksek sayıarda görülmüştür.
- HTTP Durum Kodları:** En sık karşılaşılan HTTP durum kodu **400 (Bad Request)** olup, toplamda 120 kez görülmüştür. Bunun yanı sıra **200 (Başarılı)** durumu da 57 kez kaydedilmiştir.
- Kullanıcı Aracıları:** Log dosyasında en sık görülen kullanıcı aracı **Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.5615.50 Safari/537.36**'dır.

Şüpheli Aktiviteler:

- **Anormal IP Aktivitesi:** **86.236.188[.]85** ve **254.198.150[.]19** IP adreslerinden çok sayıda istek yapılmış olması, bu IP'lerin yoğun trafiğe neden olduğunu göstermektedir. Bu durum, bir tarama veya saldırısı girişimi olabilir.
- **Bad Request (400) Durum Kodu:** **400** hata kodunun yüksek sıklığı, kullanıcıların yanlış veya eksik istekler gönderdiğini veya saldırısı amaçlı denemeler yapıldığını gösterebilir.

Öneriler:

- **IP Adreslerini İnceleyin:** Yoğun istek yapan IP adreslerinin kaynağını araştırın. Bu adreslerin meşru kullanıcılar olup olmadığını kontrol edin. Gerekirse bu IP'leri engellemek için güvenlik önlemleri alın.
- **Durum Kodu Analizi:** **400** hata kodlarının neden bu kadar yüksek olduğunu analiz edin. Kullanıcıların doğru istekler yapmalarını sağlamak için gerekli yönlendirmeleri yapın veya saldırısı olasılığına karşı önlemler alın.

5) acsess.log.3 Analizi

Bu log dosyası, bir web sunucusuna yapılan isteklerin ayrıntılarını içermektedir. Her kayıt, istemcinin IP adresi, istek zamanı, HTTP metod ve yolu, sunucunun yanıt kodu, yanıt boyutu, yönlendiren URL ve kullanıcı ajanı bilgilerini kapsamaktadır.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.3 | wc -l
2776
```

Resim 42: cat acsess.log.3 | wc -l

cat acsess.log.3 | wc -l komutu ile yapılan analizde, acsess.log.3 dosyasının toplam 2776 satırından olduğu belirlenmiştir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.3 | grep "ERROR"
```

Resim 43: cat acsess.log.3 | grep "ERROR"

grep "ERROR" komutu kullanılarak yapılan analiz sonucunda, belirtilen log dosyasında herhangi bir hata mesajı bulunmamıştır.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.3 | cut -d " " -f 1 | sort | uniq -c | sort -nr | head -n 15
74 192.99.244.139
48 31.220.113.224
48 23.254.164.173
42 31.187.79.201
35 173.44.194.237
33 31.220.30.157
30 89.42.237.71
26 162.252.172.138
24 5.157.42.183
24 104.144.19.69
22 216.244.81.34
20 94.23.33.25
19 208.115.125.58
19 195.154.46.135
18 89.36.65.53
```

Resim 44: cat acsess.log.3 | cut -d " " -f 1 | sort | uniq -c | sort -nr | head -n 15

cat acsess.log.3 | cut -d " " -f 1 | sort | uniq -c | sort -nr | head -n 15 komutu ile acsess.log.3 dosyasındaki IP adreslerinin tekrar sayıları analiz edilmiştir. Bu komut, IP adreslerini sıralayarak en çok tekrar edenleri azalan sırada listelemiştir.

Analiz: 192.99.244[.]139 IP adresi, 74 kez tekrar ederek en çok tekrar eden IP adresi olmuştur.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.3 | cut -d " " -f 1,7,9,10 | grep -i "login"
31.220.113.224 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
31.220.113.224 /login_form 200 10543
31.220.113.224 /login_form 200 16806
216.244.81.34 /login_form 200 11620
216.244.81.34 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form 200 11793
158.222.5.157 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
158.222.5.157 /login_form 200 10543
158.222.5.157 /login_form 200 16810
180.180.64.16 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
180.180.64.16 /login_form 200 10543
180.180.64.16 /login_form 200 16810
222.88.236.235 http://niels.basj.es/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basj.es/join_form 200 11713
222.88.236.235 http://niels.basj.es/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basj.es/join_form 200 11713
89.42.237.71 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form 200 11793
89.42.237.71 /login_form 200 11620
89.42.237.71 /login_form 200 18354
200.55.25.2 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
200.55.25.2 /login_form 200 10543
200.55.25.2 /login_form 200 16810
211.196.252.10 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
211.196.252.10 /login_form 200 10543
211.196.252.10 /login_form 200 16810
192.227.222.207 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
192.227.222.207 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
46.102.99.22 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basj.es/join_form 200 11713
46.102.99.22 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basj.es/join_form 200 11713
216.158.199.158 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
158.222.12.158 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
158.222.12.76 /login_form 200 10543
216.158.199.158 /login_form 200 16810
167.160.127.164 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
167.160.127.164 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form 200 10716
167.160.127.164 /login_form 200 10543
167.160.127.164 /login_form 200 16810
```

Resim 45: `cat acsess.log.3 | cut -d " " -f 1,7,9,10 | grep -i "login"`

cat acsess.log.3 | cut -d " " -f 1,7,9,10 | grep -i "login" komutu acsess.log.3 dosyasındaki her satırdan IP adresi, URL, HTTP durum kodu ve yanıt boyutlarını çıkarır ve "login" terimini içeren kayıtları filtreler.

Analiz: Login içeren isteklerin HTTP durum kodlarının 200 olduğu ve yanıt boyutlarının yüksek olduğu durumlar tespit edilmiştir.

```
└─(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log.3 | cut -d " " -f 9 | sort | uniq -c | sort -nr
 2277 200
 474 302
 11 403
   8 404
   2 123
   2
  1 Dragon/36.1.1.21
   1 "-"


```

Resim 46: `cat access.log.3 | cut -d " " -f 9 | sort | uniq -c | sort -nr`

`cat access.log.3 | cut -d " " -f 9 | sort | uniq -c | sort -nr` komutu kullanılarak, her satırındaki dokuzuncu alan seçilmiştir. Bu alanlardaki değerler sıralanmış, her bir değerin tekrar sayıları hesaplanmış ve değerler azalan sırada sıralanmıştır.

Yapılan analiz sonucunda HTTP durum kodları şu sıklıklarla tekrar etmiştir:

- **200 (OK)**: 2277 kez
- **302 (Found)**: 474 kez
- **403 (Forbidden)**: 11 kez. Bu durum kodu, "Erişim Yasak" anlamına gelir ve kullanıcının talep ettiği kaynağa erişim yetkisi olmadığını belirtir.
- **404 (Not Found)**: 8 kez
- **123**: 2 kez. Bu durum kodu, standart dışı bir HTTP kodunu temsil eder ve genellikle özel bir uygulama kodu veya loglama hatasından kaynaklanabilir. Daha fazla bilgi için sistem belgeleri ve yapılandırmalar gözden geçirilmelidir.
- **Boş**: 2 kez. Boş değer, genellikle log kaydında durum kodunun eksik olduğunu gösterir; bu durum yanıtın alınmaması, loglama hatası veya ağ sorunu nedeniyle olabilir.
- **Dragon/36.1.1.21**: 1 kez. Bu ifade, bir kullanıcı ajanı (user-agent) bilgisini belirtir ve genellikle belirli bir tarayıcı, bot veya yazılımın adını ve sürüm numarasını gösterir.
- **"-"**: 1 kez. Bu işaret, genellikle yanıtın alınmadığını veya durum kodunun kaydedilmediğini gösterir; yanıt yokluğu, veri eksikliği veya loglama sorunu olabilir.

Analiz: Log dosyasındaki HTTP durum kodları genel olarak sunucunun iyi çalıştığını ve çoğu isteğe başarılı yanıt verdiği göstermektedir. Ancak, bazı standart dışı durum kodları ve boşluklar, loglama sisteminde veya uygulama yapılandırmasında dikkat edilmesi gereken noktalar olabileceğini işaret eder. Özellikle 123 kodu ve boş değerler, sistemdeki potansiyel sorunları veya eksiklikleri tespit etmek için daha fazla inceleme gerektirebilir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat access.log.3 | cut -d " " -f 1,4,6,7,8,9,10 | grep "403"
212.129.17.73 [30/Oct/2015:12:29:20 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
183.203.23.135 [30/Oct/2015:12:30:00 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
60.191.163.235 [30/Oct/2015:10:11:21 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.0" 403 340
192.3.242.26 [30/Oct/2015:09:51:37 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
115.231.162.216 [25/Oct/2015:13:18:29 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
91.139.172.39 [29/Oct/2015:14:10:46 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
162.252.172.138 [29/Oct/2015:18:17:22 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
107.158.89.87 [30/Oct/2015:00:11:06 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
208.115.125.58 [30/Oct/2015:00:49:33 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
162.252.172.138 [29/Oct/2015:07:20:10 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
173.232.116.137 [27/Oct/2015:17:59:36 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form HTTP/1.1" 403 340
```

Resim 47: cat access.log.3 | cut -d " " -f 1,4,6,7,8,9,10 | grep "403"

cat access.log.3 | cut -d " " -f 1,4,6,7,8,9,10 | grep "403"

Komutu ile HTTP durum kodu 403 kayıtlarının analizi gerçekleştirilmiştir. İnceleme sonuçlarında aşağıdaki noktalar öne çıkmaktadır:

- Kısıtlı Erişim:** Tüm kayıtlar **403 Forbidden** hatası ile sonuçlanmıştır. Bu da URL'nin erişim kısıtlamaları nedeniyle kullanıcıların bu sayfaya erişemediklerini göstermektedir.
- Potansiyel Güvenlik Sorunu:** Aynı URL'ye farklı IP adreslerinden yapılan başarısız giriş denemeleri, bu URL'nin kötüye kullanım amacıyla hedef alınabileceğini veya brute force saldırısına maruz kalabileceğini işaret etmektedir.
- Erişim Denemeleri:** Farklı tarihlerde yapılan bu istekler, belirli bir süre boyunca düzenli olarak bu sayfaya erişim sağlanmaya çalışıldığını göstermektedir.
- HTTP Versiyonları:** İstekler hem **HTTP/1.0** hem de **HTTP/1.1** protokollerini kullanarak yapılmıştır. Bu durum, çeşitli tarayıcılar veya araçlar tarafından yapılan erişim denemelerini düşündürebilir.
- Her yanıtın boyutu **340 bayt** olarak belirtilmiştir. Yanıt boyutlarının **340 bayt** olarak sabit olması, genellikle aynı hata mesajı veya yanıt şablonunun kullanıldığını ve içerik boyutunun değişmediğini gösterir. Bu durum, yanıtların standart bir formatta olduğunu ve belirli bir hata mesajının sürekli olarak aynı boyutta sunulduğunu işaret eder.

Analiz: Bu bulgular, erişim kontrollerinin ve güvenlik önlemlerinin gözden geçirilmesi gerektiğini ortaya koymaktadır. Potansiyel güvenlik açılarını önlemek için detaylı bir inceleme ve gerekli önlemlerin alınması önerilmektedir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.3 | cut -d " " -f 7
/linux/doing-pxe-without-dhcp-control
/join_form
/join_form
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form
/login_form
/login_form
/login_form
/join_form
/join_form
/join_form
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form
/contact-info
/join_form
/join_form
/linux/doing-pxe-without-dhcp-control
/join_form
/join_form
/join_form
/join_form
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form
```

Resim 48: cat acsess.log.3 | cut -d ““ -f 7

cat access.log.3 | cut -d " " -f 7 komutu ile 7. sütundaki URL bilgileri incelendiğinde, URL encoding işlemi yapılmış bir veri dikkat çekti.

```
/linux/installing-my-new-server/networking  
join_form  
join_form  
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form  
/process?data=%3C!DOCTYPE%20root%20%5B%20%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A%2F%2Fetc%2Fshadow%22%3E%20%5D%3E%20%3Croot%3E%20%26xxe%3B%20%3C%2Froot%3E  
/process?data=%3C%2Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3F%3E%3C!DOCTYPE%20test%20%5B%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A%2F%2Fshadow%22%3E  
200  
/process?data=%3C%2Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3F%3E%3Ctest%3E%3C!%5BCDATA%5B%3C%2Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3E  
down%22%3E%D3%3Ctest%3E%26xxe%3B%3C!test%3E%5D%50%3E%3C!test%3E%20HTP%1.  
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form
```

Resim 49: `cat access.log.3 | cut -d " " -f 7`

Decode işlemi sonucunda elde edilen içerik aşağıdaki gibidir:

```
/process?data=%3C!DOCTYPE%20root%20%5B%20%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A%2F%2F%2Etc%2Fshadow%22%3E%20%5D%3E%20%3Croot%3E%20%26xxe%3B%20%3C%2Froot%3E
/process?data=%3C%3Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3F%3E%3C!DOCTYPE%20test%20%5B%20%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A//etc/shadow%22%3E%5D%3Ctest%3E%26xxe%3B%3C/test%3E%20HTTP/1.1"
```

|

① For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
/process?data=<!DOCTYPE root [ <!ENTITY xxe SYSTEM "file:///etc/shadow" > ]> <root> &xxe; </root>/process?data=<%xml version="1.0" encoding="UTF-8"?><!DOCTYPE test [<!ENTITY xxe SYSTEM "file:///etc/shadow">]><test>&xxe;</test> HTTP/1.1"
```

Resim 50: URL Decode

Derinlemesine analiz yapmak amacıyla **grep "shadow"** komutu kullanılarak arama gerçekleştirılmıştır.

```
(kali㉿kali)-[~/Desktop/logs] $ cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "shadow"
Decodes your data into the area below.
94.23.33.25 [28/Oct/2015:11:38:15 "GET /process?data=%3C!DOCTYPE%20root%20%5B%20%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A%2F%2Fetc%2Fshadow%22%3E%20%5D%3E%20%3Croot%3E%20%26xxe%3B%20%3C%2Froot%3E HTTP/1.1" 200 123
94.23.33.25 [28/Oct/2015:11:38:16 "GET /process?data=%3C%3Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3F%3C!DOCTYPE%20test%20%5B%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A///etc/%shadow%22%3E%5D%3Ctest%3E%26xxe%3B%3C/test%3E%20HTTP/1.1" 200 123
94.23.33.25 [28/Oct/2015:11:38:17 "GET /process?data=%3C%3Fxml%20version%3D%221.0%22%20encoding%3D%22UTF-8%22%3F%3C!DOCTYPE%20test%20%5B%3C!ENTITY%20xxe%20SYSTEM%20%22file%3A///etc/%shadow%22%3E%5D%3E%3Ctest%3E%26xxe%3B%3C/test%3E%5D%5D%3E%3C/test%3E%20HTTP/1.1" 200 123
```

Resim 51: `cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "shadow"`

Log kayıtlarında belirlenen veriler doğrultusunda gerçekleştirilen derinlemesine analiz ve bu verilerle ilgili olası güvenlik riskleri değerlendirilmiştir.

Analiz ve Olası Riskler:

1. XXE (XML External Entity) Saldırısı:

- **İstekler:** URL'lerde görülen istekler, XML dışa veri (XXE) saldırısını denemektedir. Bu istekler, `file[:]/etc/shadow` dosyasına erişim sağlamayı amaçlamaktadır. `/etc/shadow` dosyası, sistemdeki kullanıcı şifrelerinin hash'lerini içeren kritik bir dosyadır, bu nedenle bu tür istekler güvenlik riski taşır.

2. Kodlama ve Yönlendirme:

- **Kodlama:** URL'erdeki özel karakterlerin URL encode edilmesi, saldırının gizlenmesini ve hedef dosyaların içeriğine erişim sağlama girişimini göstermektedir.
- **Yönlendirme:** HTTP 200 durum kodları, isteklerin başarılı bir şekilde işlendiğini ve XML verilerinin hedef dosyanın içeriğiyle birlikte işlenmiş olabileceğini belirtir.

3. Riskler:

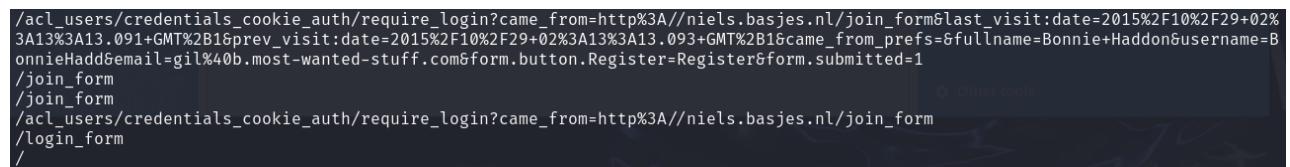
- **Güvenlik Açığı:** XML işleme mekanizmasındaki güvenlik açıkları, saldırıcıların sistem dosyalarına erişim sağlayarak hassas bilgileri çalmaya çalışabileceğini gösterir. Bu tür istekler, potansiyel veri sizıntılarına ve sistem saldırılara yol açabilir.

- **İzleme ve Önlem:** Bu tür aktivitelerin izlenmesi ve XML işleme güvenliğinin artırılması, veri sızıntılarını ve saldırıları önlemek için kritik öneme sahiptir.

4. Senkronize Saldırı Girişimleri:

- Aynı IP adresinden, aynı tarih ve saatte milisaniyelerle gerçekleştirilen tekrar eden istekler, otomatikleştirilmiş bir saldırısı veya test aracını işaret edebilir. Bu, sistem üzerinde ayrıntılı bir güvenlik testi yapıldığını ve hassas bir zamanlama kullandığını gösterir. Güvenlik önlemlerinin güçlendirilmesi ve bu tür saldırıların etkilerinin azaltılması önerilir.

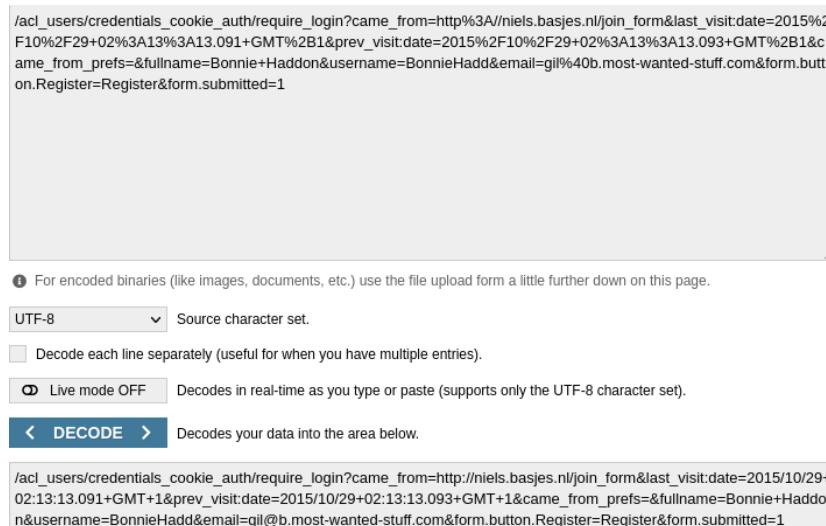
Analiz: Log kayıtlarında görülen istekler, sistemdeki kritik dosyalara erişim sağlamayı hedefleyen XXE (XML External Entity) saldırısı girişimlerini işaret etmektedir. Bu tür saldırılar, sistem güvenliğini tehlikeye atabilir ve hassas bilgilerin sızmasına neden olabilir. Güvenlik açıklarının değerlendirilmesi ve uygun önlemlerin alınması kritik öneme sahiptir.



```
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form&last_visit:date=2015%2F10%2F29+02%3A13%3A13.091+GMT%2B1&prev_visit:date=2015%2F10%2F29+02%3A13%3A13.093+GMT%2B1&came_from_prefs=&fullname=Bonnie+Haddon&username=BonnieHadd&email=gil%40b.most-wanted-stuff.com&form.button.Register=Register&form.submitted=1
/join_form
/join_form
/acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form
/login_form
/
```

Resim 52: cat acsess.log.3 | cut -d “ “ -f 7

cat acsess.log.3 | cut -d “ “ -f 7 komutu ile sütun 7’deki verileri incelerken, ‘username’ ve ‘email’ gibi kişisel bilgileri içeren bir satır dikkat çekmiştir.



The screenshot shows a URL decode interface with the following details:

- Input text area: /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form&last_visit:date=2015%2F10%2F29+02%3A13%3A13.091+GMT%2B1&prev_visit:date=2015%2F10%2F29+02%3A13%3A13.093+GMT%2B1&came_from_prefs=&fullname=Bonnie+Haddon&username=BonnieHadd&email=gil%40b.most-wanted-stuff.com&form.button.Register=Register&form.submitted=1
- Character set dropdown: UTF-8
- Decode options: "Decode each line separately (useful for when you have multiple entries)" is checked.
- Live mode: "Live mode OFF" is selected.
- Decode button: "DECODE" with a left arrow and right arrow icon.
- Output area: /acl_users/credentials_cookie_auth/require_login?came_from=http://niels.basjes.nl/join_form&last_visit:date=2015/10/29+02:13:13.091+GMT+1&prev_visit:date=2015/10/29+02:13:13.093+GMT+1&came_from_prefs=&fullname=Bonnie+Hadd&username=BonnieHadd&email=gil@b.most-wanted-stuff.com&form.button.Register=Register&form.submitted=1

Resim 53: URL Decode

Veri decode edilerek daha anlaşılır bir şekilde detaylı inceleme yapılmıştır.

URL İçeriği:

- **came_from=hxxp://niels.basjes[.]nl/join_form:** Kullanıcının giriş işlemi tamamlandıktan sonra yönlendirileceği URL'yi belirtir. Başarılı bir giriş sonrası kullanıcı, http://niels.basjes.nl/join_form adresine yönlendirilir.
- **last_visit=2015/10/29+02:13:13.091+GMT+1:** Kullanıcının son ziyaretinin tarihi ve saati.
- **prev_visit=2015/10/29+02:13:13.093+GMT+1:** Kullanıcının önceki ziyaretinin tarihi ve saati.
- **came_from_prefs=:** Kullanıcının giriş yapmadan önceki tercihlerle ilgili bir parametre olup, burada boş bırakılmış.
- **fullname=Bonnie+Haddon:** Kullanıcının tam adı (Bonnie Haddon).
- **username=BonnieHadd:** Kullanıcının kullanıcı adı.
- **email=gil@b.most-wanted-stuff.com:** Kullanıcının e-posta adresi.
- **form.button.Register=Register:** Formun kayıt amacıyla gönderildiğini belirten parametre.
- **form.submitted=1:** Formun başarıyla gönderildiğini gösterir.

Analiz: URL içerisinde, kullanıcı adı, e-posta adresi gibi kişisel bilgilerin yanı sıra giriş ve kayıt işlemleriyle ilgili veriler bulunmaktadır. Bu verilerin URL'de açıkça görünmesi, kişisel bilgilerin ifşa olmasına ve güvenlik risklerine yol açabilir. URL üzerinde kişisel bilgilerin taşınması, potansiyel veri sizıntısı ve kimlik doğrulama açıklarına işaret eder. Güvenlik açıklarının önlenmesi için kişisel bilgilerin URL'lerde yerine güvenli yollarla iletilmesi ve ek güvenlik önlemleri alınması gerekmektedir.

```
(kali㉿kali)-[~/Desktop/logs]
└─$ cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "email"
103.27.239.39 [25/Oct/2015:10:52:44 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl
/join_form&last_visit:date=2015%2F10%2F13+18%3A31%3A02.492+GMT%2B2&prev_visit:date=2015%2F10%2F13+18%3A31%3A02.493+GMT%2B2&c
ame_from_prefs=&fullname=Eileen+Coe&username=EileenCoe&email=t.h.u.c.d.v2016%40gmail.com&form.button.Register=Register&form.
submitted=1 HTTP/1.1" 200
117.169.6.110 [29/Oct/2015:02:09:06 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl
/join_form&last_visit:date=2015%2F10%2F29+02%3A13%3A13.091+GMT%2B1&prev_visit:date=2015%2F10%2F29+02%3A13%3A13.093+GMT%2B1&c
ame_from_prefs=&fullname=Bonnie+Haddon&username=BonnieHadd&email=gil%40b.most-wanted-stuff.com&form.button.Register=Register
&form.submitted=1 HTTP/1.1" 200
```

Resim 54: cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "email"

E-posta adreslerini içeren satırları incelemek amacıyla **grep "email"** komutu kullanıldı. Bu işlem sonucunda benzer bir satır daha tespit edildi ve bu veriye aynı analiz uygulandı.

The screenshot shows a URL decoding interface. At the top, there is a text input field containing the following URL:

```
103.27.239.39 [25/Oct/2015:10:52:44 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niel
s.basjes.nl/join_form&last_visit:date=2015%2F10%2F13+18%3A31%3A02.492+GMT%2B2&prev_visit:date=2015%2F1
0%2F13+18%3A31%3A02.493+GMT%2B2&came_from_prefs=&fullname=Eileen+Coe&username=EileenCoe&email=t.h.
u.c.d.v2016%40gmail.com&form.button.Register=Register&form.submitted=1 HTTP/1.1" 200
```

Below the input field are several configuration options:

- UTF-8** dropdown: Source character set.
- Decode each line separately (useful for when you have multiple entries).
- Live mode OFF: Decodes in real-time as you type or paste (supports only the UTF-8 character set).
- DECODE** button: Decodes your data into the area below.

The output area displays the decoded URL:

```
103.27.239.39[25/Oct/2015:10:52:44"GET/acl_users/credentials_cookie_auth/require_login?came_from=http://niels.basje
s.nl/join_form&last_visit:date=2015/10/13+18:31:02.492+GMT+2&prev_visit:date=2015/10/13+18:31:02.493+GMT+2&ca
me_from_prefs=&fullname=Eileen+Coe&username=EileenCoe&email=t.h.u.c.d.v2016@gmail.com&form.button.Register=
Register&form.submitted=1HTTP/1.1"200
```

Resim 55: URL Decode

Bu URL içeriği, kişisel kullanıcı bilgilerini ve form gönderim detaylarını içerdığı için güvenlik açısından dikkate alınması gereken potansiyel riskler taşımaktadır.

Analiz: Decode edilmiş URL verileri, önemli kişisel bilgileri içeren giriş ve kayıt formu verilerini göstermektedir. URL'deki özel karakterlerin URL encode edilmesi, verilerin gizlenmeye çalışıldığını ve bu bilgilerin potansiyel olarak kötüye kullanılabileceğini işaret etmektedir. HTTP 200 durum kodu, bu isteklerin başarılı bir şekilde işlendiğini ve formun başarıyla gönderildiğini doğrular.

Bu veriler, kişisel bilgilerin (kullanıcı adı, e-posta adresi gibi) ifşasının kullanıcı güvenliği açısından ciddi bir tehdit oluşturabileceğini ve kimlik doğrulama ile form işleme süreçlerinde potansiyel güvenlik açıklarını işaret ettiğini ortaya koymaktadır. Bu durum, veri sizintisi ve diğer güvenlik ihlallerine yol açabilir.

Ek güvenlik önlemleri ve sürekli izleme gereklidir. Kullanıcı bilgileri güvenli bir şekilde korunmalı ve kimlik doğrulama süreçleri güçlendirilmelidir. Bu tür güvenlik açıklarının önlenmesi, veri güvenliği ve kullanıcı gizliliğinin sağlanması açısından kritik öneme sahiptir.

```
(kali㉿kali)-[~/Desktop/logs]
$ cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "103.27.239.39"
103.27.239.39 [25/Oct/2015:10:52:44 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl
/join_form&last_visit:date=2015%2F10%2F13+18%3A31%3A02.492+GMT%2B2&prev_visit:date=2015%2F10%2F13+18%3A31%3A02.493+GMT%2B2&c
ame_from_prefs=&fullname=Eileen+Coe&username=EileenCoe&email=t.h.u.c.d.v2016%4@gmail.com&form.button.Register=Register&form.
submitted=1 HTTP/1.1" 200
103.27.239.39 [25/Oct/2015:10:52:46 "GET /join_form HTTP/1.1" 200
103.27.239.39 [25/Oct/2015:10:52:48 "POST /join_form HTTP/1.1" 302
103.27.239.39 [25/Oct/2015:10:52:50 "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl
/join_form HTTP/1.1" 200
103.27.239.39 [25/Oct/2015:10:52:51 "GET /login_form HTTP/1.1" 200
103.27.239.39 [25/Oct/2015:10:52:54 "POST /login_form HTTP/1.1" 200
```

Resim 56: cat acsess.log.3 | cut -d " " -f 1,4,6,7,8,9 | grep "103.27.239.39"

103.27.239.39 IP adresinin hareketleri daha detaylı incelenmiştir.

Analiz: 103.27.2398[.]39 IP adresine ait hareketler, 25 Ekim 2015 tarihinde kısa aralıklarla yapılan ve kişisel bilgileri içeren formların gönderildiği bir dizi isteği göstermektedir. Bu istekler, belirli sayfalara erişim sağlamak amacıyla gerçekleştirilmiş ve özellikle arka arkaya yapılan istekler, otomatikleştirilmiş bir aracın veya botun kullanımına işaret etmektedir. Bu tür aktiviteler, sistemdeki güvenlik açıklarını test etmeyi amaçlayan bir saldırı veya kötüye kullanım girişimi olabilir.

Bu gözlemler, kişisel verilerin korunması ve sistem güvenliğinin artırılması gerektiğini vurgular. Otomatikleştirilmiş erişimlerin ve potansiyel saldırıların önlenmesi için güvenlik önlemlerinin güçlendirilmesi, veri güvenliği ve kullanıcı gizliliği açısından kritik öneme sahiptir.

```
(kali㉿kali)-[~/Desktop/logs] $ cat acsess.log.3 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 15
944 /join_form
810 /login_form
354 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form
235 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basjes.nl/join_form
88 /
75 /linux/doing-pxe-without-dhcp-control
67 /linux/installing-fedora-linux-via-pxe-x86-64
39 /places
25 /open-source
23 /linux/installing-my-new-server/networking
22 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//niels.basj.es/join_form
11 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//daniel_en_sander.basjes.nl/join_form
11 /accessibility-info
9 /linux/installing-my-new-server/vmware-server
9 /linux/installing-gitlab-on-centos-6
```

Resim 57: `cat acsess.log.3 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 1`

En fazla erişim sağlanan URL'leri belirlemek amacıyla `cat acsess.log.3 | cut -d " " -f 7 | sort | uniq -c | sort -nr | head -n 15` komutu kullanılmıştır.

Analiz: Bu analiz sonucunda, en sık erişilen URL'nin **/join_form** olduğu tespit edilmiştir. Ayrıca bazı URL'lere belirgin bir şekilde daha fazla erişim yapıldığı ve bazı URL'lerin ise daha az sıklıkta ziyaret edildiği anlaşılmıştır.. Özellikle **join_form** ve **login_form** gibi URL'lerin yüksek erişim sayıları, bu sayfalara olan ilginin yoğun olduğunu veya bu URL'lerin potansiyel olarak hedef alınmış olabileceği düşünürebilir.

```
(kali㉿kali)-[~/Desktop/logs] $ cat acsess.log.3 | cut -d " " -f 1,7 | grep "/join_form" | sort | uniq -c | sort -nr | head -n 5
22 192.99.244.139 /join_form
16 31.220.113.224 /join_form
16 23.254.164.173 /join_form
14 31.187.79.201 /join_form
13 94.23.33.25 /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form

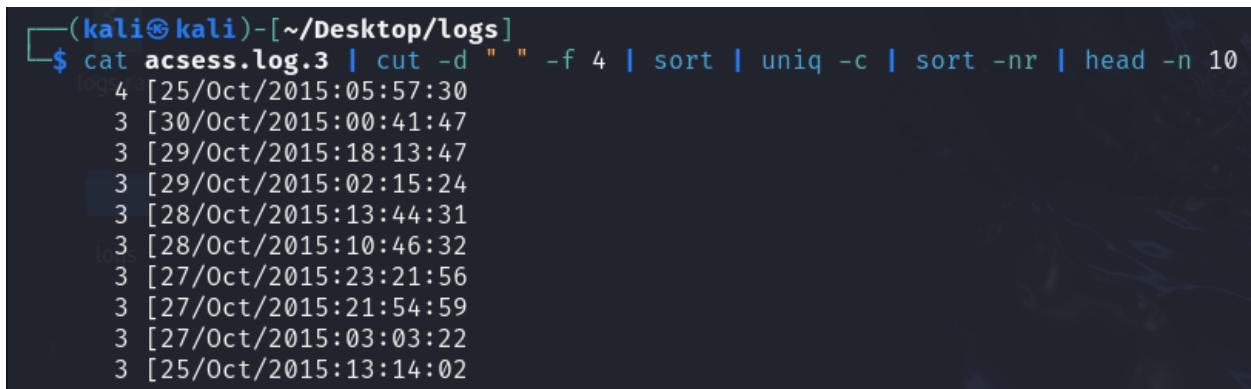
(kali㉿kali)-[~/Desktop/logs] $ cat acsess.log.3 | cut -d " " -f 1,7 | grep "/login_form" | sort | uniq -c | sort -nr | head -n 5
22 192.99.244.139 /login_form
16 31.220.113.224 /login_form
16 23.254.164.173 /login_form
14 31.187.79.201 /login_form
10 89.42.237.71 /login_form
```

Resim 58: `cat acsess.log.3 | cut -d " " -f 1,7 | grep "/join_form" | sort | uniq -c | sort -nr | head -n 5 ve grep "/login_form"`

`cat acsess.log.3 | cut -d " " -f 1,7 | grep "/join_form" | sort | uniq -c | sort -nr | head -n 5` komutu ve `grep "/login_form"` komutu kullanılarak bu URL'lere en sık erişim sağlayan IP adresleri tespit edilmiştir.

Analiz: Log dosyasındaki `/join_form` ve `/login_form` yollarına yapılan isteklerin analizi, her iki URL için de benzer IP adreslerinden gelen yüksek frekansta isteklerin olduğunu ortaya koymuştur. Bu durum, aynı IP adreslerinin her iki kritik endpoint'e yönelik tekrarlayan isteklerde bulunduğu göstermektedir.

Bu tür bir davranış, muhtemel bir brute-force saldırısı veya kimlik doğrulama bypass girişimi olarak değerlendirilebilir. Özellikle, iki farklı kritik endpoint'e yönelik sürekli olarak yapılan bu istekler, otomatikleştirilmiş araçların veya botların hedef alabileceği potansiyel bir saldırı girişimini işaret edebilir. Bu, sistemin kimlik doğrulama ve erişim kontrol mekanizmalarının gözden geçirilmesi gerektiğini ve ek güvenlik önlemleri alınması gerektiğini göstermektedir.



```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.3 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10
4 [25/Oct/2015:05:57:30
 3 [30/Oct/2015:00:41:47
 3 [29/Oct/2015:18:13:47
 3 [29/Oct/2015:02:15:24
 3 [28/Oct/2015:13:44:31
 3 [28/Oct/2015:10:46:32
 3 [27/Oct/2015:23:21:56
 3 [27/Oct/2015:21:54:59
 3 [27/Oct/2015:03:03:22
 3 [25/Oct/2015:13:14:02
```

Resim 59: `cat access.log.3 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10`

Tarih ve saat verilerini analiz etmek amacıyla `cat access.log.3 | cut -d " " -f 4 | sort | uniq -c | sort -nr | head -n 10` komutu kullanılmıştır.

Analiz: Bu analiz sonucunda, en sık görülen tarih ve saat `'25/Oct/2015:05:57:30'` olarak tespit edilmiştir.



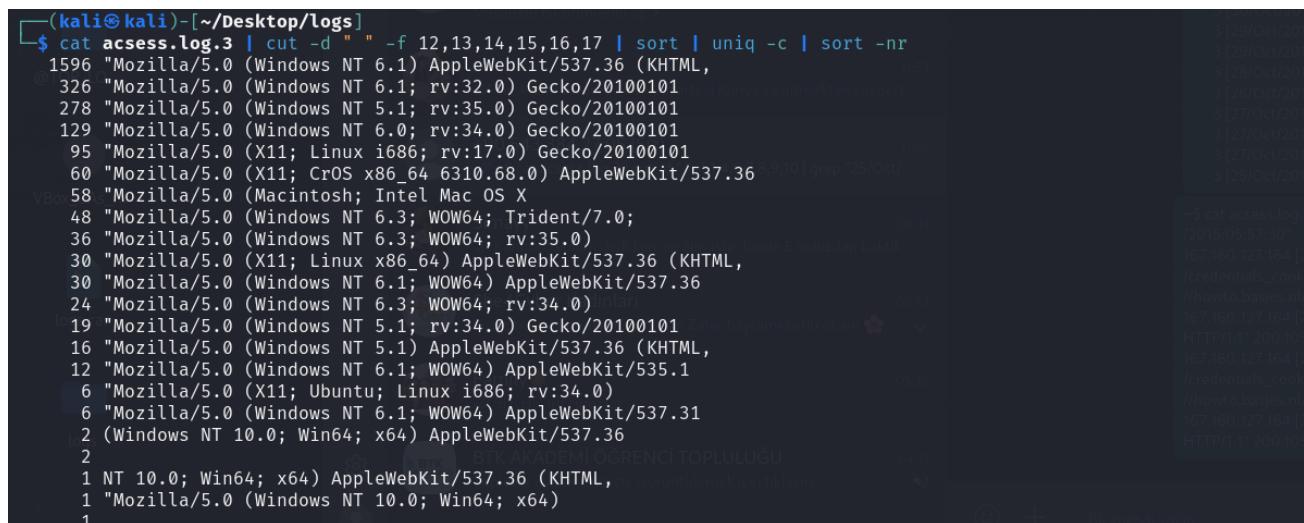
```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.3 | cut -d " " -f 1,4,6,7,8,9,10 | grep "25/Oct/2015:05:57:30"
167.160.127.164 [25/Oct/2015:05:57:30] "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form HTTP/1.1" 200 10716
167.160.127.164 [25/Oct/2015:05:57:30] "GET /login_form HTTP/1.1" 200 10543
167.160.127.164 [25/Oct/2015:05:57:30] "GET /acl_users/credentials_cookie_auth/require_login?came_from=http%3A//howto.basjes.nl/join_form HTTP/1.1" 200 10716
167.160.127.164 [25/Oct/2015:05:57:30] "GET /login_form HTTP/1.1" 200 10543
```

Resim 60: `cat access.log.3 | cut -d " " -f 1,4,6,7,8,9,10 | grep "25/Oct/2015:05:57:30"`

En sık tekrarlanan tarihi incelemek amacıyla `grep "25/Oct/2015:05:57:30"` komutu kullanılmıştır. Bu tarih için yapılan analizde elde edilen bulgular aşağıdaki gibidir:

- Aynı IP Adresinden Tekrar Eden İstekler:** IP adresi 167.160.127.164 tarafından aynı anda iki farklı URL'ye (/acl_users/credentials_cookie_auth/require_login ve /login_form) tekrar eden istekler yapılmıştır.
- Başarıyla Yanıtlanması:** Her iki istek de başarılı bir şekilde yanıtlanmış ve yanıt boyutları birbirine yakın olmuştur.
- Olası Otomatik Araç Kullanımı:** Aynı anda birden fazla istek yapılması, otomatik bir araç veya bot kullanılarak gerçekleştirilen testleri veya potansiyel bir saldırıyı işaret edebilir.

Analiz: Bu bulgular, sistemdeki anormal trafik desenlerini ve potansiyel güvenlik risklerini belirlemek açısından önemlidir. Bu tür aktivitelerin izlenmesi ve güvenlik önlemlerinin güçlendirilmesi, sistem koruma ve olası saldırırlara karşı hazırlıklı olma açısından kritik öneme sahiptir.



```
(kali㉿kali)-[~/Desktop/logs]
$ cat access.log.3 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr
1596 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4369.106 Safari/537.36" [29/Okt/2018:10:31:31 +0200] 3 [29/Okt/2018:10:31:31 +0200]
326 "Mozilla/5.0 (Windows NT 6.1; rv:32.0) Gecko/20100101 Firefox/32.0" [28/Okt/2018:10:31:31 +0200] 3 [28/Okt/2018:10:31:31 +0200]
278 "Mozilla/5.0 (Windows NT 5.1; rv:35.0) Gecko/20100101 Firefox/35.0" [27/Okt/2018:10:31:31 +0200] 3 [27/Okt/2018:10:31:31 +0200]
129 "Mozilla/5.0 (Windows NT 6.0; rv:34.0) Gecko/20100101 Firefox/34.0" [27/Okt/2018:10:31:31 +0200] 3 [27/Okt/2018:10:31:31 +0200]
95 "Mozilla/5.0 (X11; Linux i686; rv:17.0) Gecko/20100101 Firefox/17.0" [27/Okt/2018:10:31:31 +0200] 3 [27/Okt/2018:10:31:31 +0200]
60 "Mozilla/5.0 (X11; CrOS x86_64 6310.68.0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
58 "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:35.0) Gecko/20100101 Firefox/35.0" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
48 "Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) Microsoft Internet Explorer/11.0.9600.17500" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
36 "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:35.0) Gecko/20100101 Firefox/35.0" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
30 "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
30 "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
24 "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:34.0) Gecko/20100101 Firefox/34.0" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
19 "Mozilla/5.0 (Windows NT 5.1; rv:34.0) Gecko/20100101 Firefox/34.0" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
16 "Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
12 "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko) Chrome/14.0.835.47 Safari/535.1" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
6 "Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:34.0) Gecko/20100101 Firefox/34.0" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
6 "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.31 (KHTML, like Gecko) Chrome/31.0.1650.63 Safari/537.31" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
2 "(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
1 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
1 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.101 Safari/537.36" [25/Okt/2018:10:31:31 +0200] 3 [25/Okt/2018:10:31:31 +0200]
```

Resim 61: cat access.log.3 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr

cat access.log.3 | cut -d " " -f 12,13,14,15,16,17 | sort | uniq -c | sort -nr kullanılarak, web sunucusuna yapılan isteklerdeki User-Agent başlıklarını analiz edilmiştir.

Analiz Sonuçları:

- En Yaygın Kullanıcı Ajanı: Windows NT 6.1 tabanlı ve AppleWebKit/537.36 içeren kullanıcı ajanı, 1596 kez görünmüştür. Bu, en sık kullanılan tarayıcı ve işletim sistemi kombinasyonunu göstermektedir.

- Diğer Yaygın Ajanlar: Windows NT 6.1; rv:32.0 ve Windows NT 5.1; rv:35.0 tabanlı kullanıcı ajanları sırasıyla 326 ve 278 kez gözlemlenmiştir. Bu ajanlar da oldukça yaygın olarak kullanılmaktadır.
- Nadir Ajanlar: Windows NT 10.0 tabanlı ve Ubuntu tabanlı kullanıcı ajanları daha az sıklıkta gözlemlenmiştir, sırasıyla sadece birkaç kez görülmüştür.

Bu analiz, farklı tarayıcı ve işletim sistemlerinin ne sıklıkla kullanıldığını belirlemeye yardımcı olmuştur.

“acsess.log.3” Dosyası Üzerinde Şüpheli Aktivitelerin Genel Değerlendirilmesi

1. İstemci IP Adresleri

Log dosyasındaki bazı IP adreslerinin tekrarlanan istekleri, kötü niyetli aktivitelerin işaretini olabilir. Özellikle /login_form ve /join_form gibi sayfalara yapılan POST istekleri, kullanıcı kimlik doğrulama sistemlerini hedef almış olabilir.

Bu durum, kısa aralıklarla yapılan tekrarlanan isteklerin, otomatik araçlar veya botlar tarafından gerçekleştirilen kötü niyetli faaliyetleri işaret ettiğini gösterir. Bu tür IP adresleri, şifre kırma girişimlerinde veya diğer saldırı türlerinde yer alıyor olabilir.

2. HTTP Yöntemleri

- **GET İstekleri:** Sayfa yüklemeleri ve veri alımı için yaygın olarak kullanılır ve genellikle normal kullanıcı davranışını temsil eder. Ancak, büyük miktarda veri çekme amacıyla yapılan GET istekleri bir saldırının parçası olabilir.
- **POST İstekleri:** Hassas işlemler için kullanılır ve tekrarlanan POST istekleri, parola tahmin etme saldırısının göstergesi olabilir. Özellikle /login_form ve /join_form gibi kritik endpointlere yönelik tekrarlanan POST istekleri dikkatle izlenmelidir.

3. Yanıt Durum Kodları

- **200 OK:** İsteklerin başarıyla tamamlandığını gösterir. Bu durum, normal kullanıcı işlemleri yanı sıra, saldırganların başarılı bir şekilde verilere erişim sağladığını da gösterebilir.
- **302 Found:** Yönlendirmeyi belirtir. POST isteklerinden sonra bu durum kodunun görülmesi, kullanıcıların giriş yapma veya form gönderme girişimlerinde bulunmuş olabileceğini gösterir.

Bu durum kodlarının yaygınlığı, başarılı işlemler ve yönlendirmeleri işaret eder. Ancak, özellikle POST istekleriyle birlikte sık görülen 302 kodları, otomatik saldırının bir parçası olabilir.

4. Kullanıcı Aracı (User Agent)

Log dosyasındaki User Agent bilgileri, belirli tarayıcı ve işletim sistemi kombinasyonlarını göstermektedir. Özellikle belirli User Agent ile yapılan tekrarlanan istekler, otomatik araçların veya botların kullanılma olasılığını artırır. Şüpheli IP adreslerinden gelen bu tür istekler dikkatle izlenmelidir.

5. Zaman Damgaları

İsteklerin zaman damgaları, belirli aralıklarda yoğunlaşan istekler gösterir. Özellikle belirli IP adreslerinden gelen isteklerin zaman aralıkları incelenmelidir. Yoğunluk belirli aralıklarla, özellikle gece saatlerinde veya olağandışı yoğunlukta ise, bu durum sistemin zayıf noktalarının test edildiğini gösterebilir.

Sonuç ve Öneriler:

1. **Şüpheli IP Adreslerinin İzlenmesi ve Engellenmesi:** Şüpheli aktiviteler gösteren IP adresleri izlenmeli ve gerekirse geçici olarak engellenmelidir. Özellikle dikkat çekici IP adresleri üzerine odaklanılmalıdır.
2. **Rate Limiting Uygulanması:** Aynı IP adresinden kısa süre içinde yapılan çok sayıda isteği sınırlamak için rate limiting uygulanabilir. Bu, hızlı deneme yanlışlıkla yöntemleriyle yapılabilecek saldıruları engelleyebilir.
3. **POST İsteklerine Özel Güvenlik Kontrolleri:** POST istekleri için ek güvenlik kontrolleri uygulanabilir. CAPTCHA kullanımı ve başarılı POST isteklerinden sonra belirli bir süre yeni isteklerin engellenmesi gibi önlemler düşünülebilir.

4. **Log Analiz Araçları Kullanımı:** Log analizini daha otomatik hale getirmek için araçlar (örneğin, Splunk, ELK Stack) kullanılabilir. Bu araçlar, şüpheli aktiviteleri otomatik olarak tespit etmek ve raporlamak için yapılandırılabilir.
5. **Kullanıcı Bilgilendirmesi ve Eğitimi:** Kullanıcıların güçlü parolalar kullanmaları ve şüpheli aktiviteler konusunda dikkatli olmaları sağlanmalıdır. Güçlü parola politikaları uygulanmalı ve kullanıcıların güvenlik bilinci artırılmalıdır.

KAYNAKÇA

<https://adlibilisimhizmetleri.com/log-nedir/>

<https://chatgpt.com/>