

LINUX

FORENSIC

İçindekiler Tablosu

İÇİNDEKİLER TABLOSU	2
ÖNSÖZ	3
LINUX DİZİN YAPISI	4
HAZIRLAYAN	12

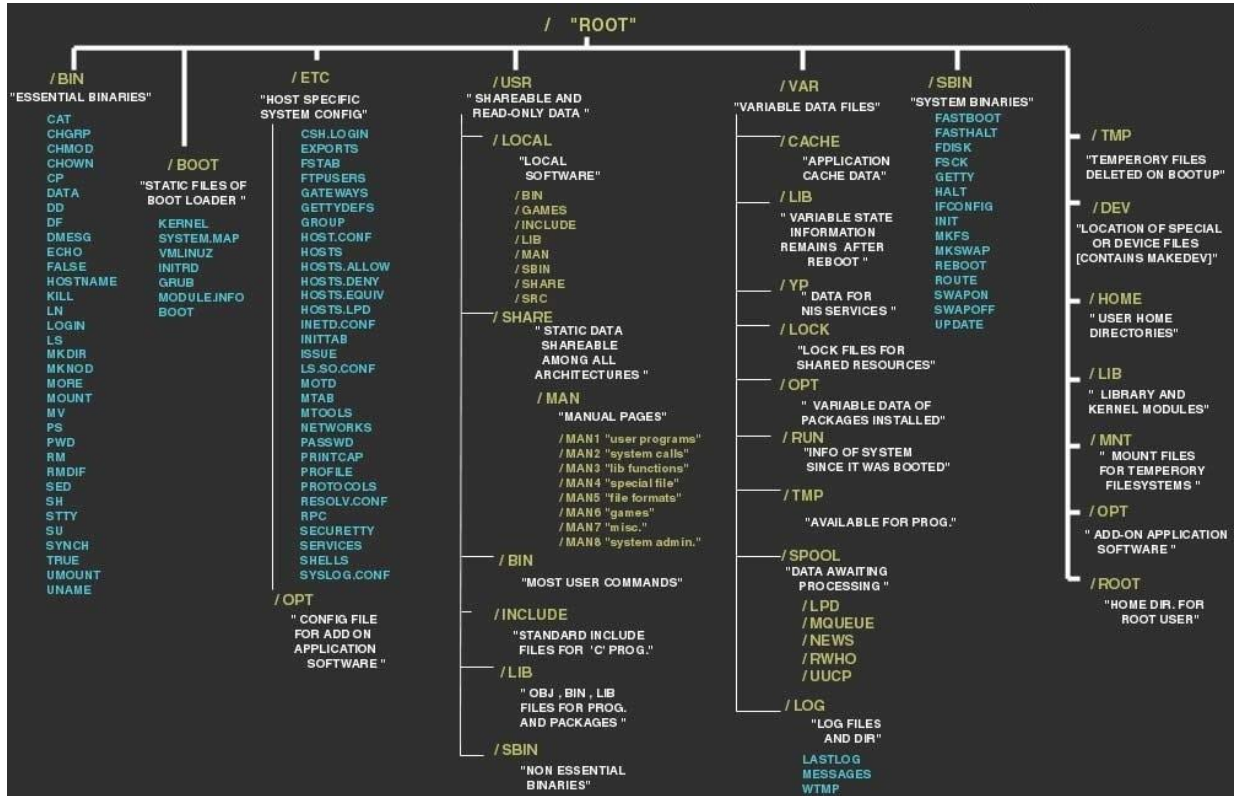
ÖN SÖZ

Linux forensic, sistem olaylarını analiz ederek dijital suçlarla ilgili kanıt toplama sürecidir. Bu süreçte sistem logları, dosya yapıları, kullanıcı etkinlikleri ve ağ bağlantıları gibi veriler incelenir. Örneğin, `/var/log/` dizinindeki log dosyaları, kullanıcı aktiviteleri ve başarısız giriş denemeleri gibi kritik bilgileri içerir. SSH logları ve cron job'lar, uzaktan erişim ve zamanlanmış görevlerin analizi için kullanılırken, paket yönetim logları izinsiz kurulumları tespit etmeye yardımcı olur.

Adli inceleme sırasında, sistemin dikkatle yedeklenmesi ve güçlü Linux araçlarının (örn. `journalctl`, `netstat`) kullanımı önemlidir. Linux'un esnek yapısı, adli incelemeyi kolaylaştırırken aynı zamanda uzmanlık gerektirir. Bu alan, hem suçların aydınlatılmasında hem de güvenlik açıklarının kapatılmasında kritik bir rol oynar.

Linux adlı bilişim süreci, Linux sistemlerinde depolanan verilerin incelenmesi, analiz edilmesi ve suç teşkil eden ya da olayın aydınlatılmasına yardımcı olabilecek kanıtların toplanmasıyla ilgilidir. Sistem günlükleri, kullanıcı etkinlikleri, ağ trafiği ve dosya sistemleri gibi bir dizi bileşen dikkatlice incelenerek, olayın nasıl gerçekleştiği ve kimlerin sorumlu olduğunu belirlenmeye çalışılır. Linux'un açık kaynak yapısı, güçlü analiz araçları sunarken iz bırakmadan kötü niyetli faaliyetlere de olanak tanıyabilir, bu nedenle adli süreç dikkat ve uzmanlık gerektirir.

LINUX DIZIN YAPISI



Linux'da tüm veriler birer dizin olarak görünür. Diskteki verilerden tutunda sisteme takılan hoparlör gibi aygıtlara kadar. Yukarıda bu dizinlerin bir şeması verilmiştir. Bu dizinleri tek tek açıklamak gerekirse:

/ (Root-Kök): Diğer tüm dizinleri kapsayan, en büyük ebeveyn dizin / (root) dizinidir. Altında 15-16 dizin bulunur. / (root) dizininin silinmesi tüm sisteminizin silinmesi demektir.

/bin: Binary yazılımının bir kısaltmasıdır. Burada sistemde çalıştırılan komutların derlenmiş halleri bulunur. Sistem için kritiktir. Bu dizin silinirse sistemde hiçbir komut çalıştırılmaz. Temel sistem komutları bu dizin içinde bulunur.

/boot: Kernel ve açılış dosyalarını içerir. Sistem ilk başlarken bu dosyalardan yüklenilir.

/etc: Sistemin yapılandırma dosyalarının birçoğu bu dizinde bulunur. Sistemde bir ayar yapılacağı zaman genellikle buradan yapılır.

/opt: Kendi kütüphane dosyalarını yükleyen uygulamaların kütüphane dosyalarını tutar. Silinmesi halinde bağlı uygulamalarda hata oluşabilir.

/usr: Çalıştırılabilir dosyalar, kaynak kodlar, dökümanlar, kütüphaneler gibi dosyalar içerir. Kullanıcıların uygulamaları burada depolanır. /usr/bin ve /usr/sbin dizinleri de temel sistem komutlarını içerir.

/lib: Kütüphane ve kernel modülleri tutulur.

/sbin: Sistem çalıştırılabilir dosyaları saklanır.

/var: Variable(değişken) ifadesinin kısaltılmışıdır. İçerisinde sürekli değişecek bilgileri tutar. Örneğin loglar veya mail bilgileri gibi.

/tmp: Geçici dosyaların bulunduğu kısımdır. Burada root kullanıcısı dışında her kullanıcı kendi dosyalarında değişiklik yapabilir.

/proc: Sistemle ilgili bilgilerin alınabileceği bölümdür. Aslında boştur fakat sistem çalıştırıldığında sistemle ilgili bilgiler alınabilir.

/mnt: Geçici mount edilmiş dosya sistemlerinin tutulduğu dizindir.

/media: Çıkarılabilir medya cihazlarının mount edildiği dizindir.

/home: Kullanıcıların ev dizinlerinin tutulduğu dizindir. İçindeki bir dizinin silinmesi durumunda silinen dizinin sahibi olan kullanıcı dosyalarını kaybetmiş olur.

/srv: srv services kelimesinin kısaltmasından gelir. Sistem tarafından sunulan servis dosyalarının bulunduğu dizindir.

/lost+found: Sistemde kaybolan dosyaların tutulduğu dizindir. Herhangi bir sebepten dosyanız kaybolursa bu dizine bakmakta fayda var.

Sistem hakkında bilgi toplamak için birçok alt dizin vardır. İşe ilk önce sistem hakkında bilgi toplamak ile başlanılabilir. Bunun **/etc/os-release** adresinde bulunan dosyayı cat ile okumak yardımcı olabilir. Ayrıca kernel ve sistem bilgilerini görüntülemek için **uname -a** komutu da kullanılabilir.

```
(kali㉿kali)-[~]
$ cat /etc/os-release
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
VERSION_ID="2024.2"
VERSION="2024.2"
VERSION_CODENAME=kali-rolling
ID=kali
ID_LIKE=debian
HOME_URL="https://www.kali.org/"
SUPPORT_URL="https://forums.kali.org/"
BUG_REPORT_URL="https://bugs.kali.org/"
ANSI_COLOR="1;31"

(kali㉿kali)-[~]
$ uname -a
Linux kali 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.15-2kali1 (2024-05-17) x86_64 GNU/Linux
```

Sistem mimarisi, çekirdek sürümü ve işletim sistemi gibi bilgiler, doğru yazılımların seçilmesi, güvenlik yamalarının uygulanması ve gelecekteki ihtiyaçlara yönelik planlamalar yapılmasına olanak tanır. Sürümlerine göre sistemlere uygulanacak güvenlik önlemleri farklılık gösterebilir.

/etc/passwd dosyası sistemdeki kullanıcılar hakkında bilgiler verir. Cat ile bu dosya okunabilir. Çıktıda kullanıcı adı, şifre bilgileri, kullanıcı kimliği(uid), grup kimliği(gid), kullanıcı oturum açıldığında çalıştırılan kabuk gibi bilgileri içerir. Windows'ta olduğu gibi, kullanıcı tarafından oluşturulan kullanıcı hesaplarının kullanıcı kimlikleri 1000 veya üzeridir. Grep, awk gibi komutlarla filtreleme yapılarak istenilen bilgilere daha çabuk ulaşılabilir.

```
(kali㉿kali)-[~]
$ cat /etc/passwd | column -t -s :
root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534:/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/usr/sbin/nologin
systemd-timesync:x:992:992:systemd Time Synchronization:/usr/sbin/nologin
messagebus:x:100:102:/nonexistent:/usr/sbin/nologin
tss:x:101:104:TPM software stack,,/var/lib/tpm:/bin/false
strongswan:x:102:65534:/var/lib/strongswan:/usr/sbin/nologin
tcpdump:x:103:105:/nonexistent:/usr/sbin/nologin
sshd:x:104:65534:/run/sshd:/usr/sbin/nologin
usbmux:x:105:46:usbmux daemon,,/var/lib/usbmux:/usr/sbin/nologin
```

Linux sistemlerde **group (grup)**, kullanıcıların ortak erişim izinlerini yönetmek için oluşturulan birimdir. Kullanıcılar gruplara dahil edilerek dosya ve kaynaklara erişimleri kolayca düzenlenir. Gruplar sayesinde, kullanıcılar arasında belirli bir kaynak paylaşımı, grup düzeyinde tanımlanan izinlerle kolayca sağlanır. Cat ile **/etc/group** dizini okunarak grup bilgilerine ulaşılabilir.

```
(kali㉿kali)-[~]
$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:kali
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:kali
fax:x:21:
voice:x:22:
cdrom:x:24:kali
floppy:x:25:kali
```

Örneğin fotoğrafta görüldüğü gibi adm kullanıcı grubunun gid'si 4 ve bu grupta kali kullanıcı bulunmaktadır yorumu yapılabilir.

Journalctl komutu, sistem ve servislerle ilgili tüm logları izlemek için kullanılabilir. Loglar tek bir yerde toplandığında hata teşhisi, performans ve güvenlik analizi gibi işlemler belirli bir tarih, servis veya öneme göre filtrelenerek kolaylıkla yapılabilir. Örneğin sistem boot kayıtları, servis hataları veya kernel mesajlarını incelemek fayda sağlayabilir. Journalctl ile kullanılacak komutlar şunlardır:

journalctl -b: Sistemin her açılışında oluşan logları görmek için kullanılır.

journalctl -u servis_adi: Belirtilen servisin loglarını incelemek için kullanılır. **--priority=3** parametresiyle sadece ilgili serviste oluşan hatalar görünebilir.

journalctl -k: Sadece kernel ile ilgili mesajları görüntüler. **-f** parametresiyle gerçek zamanlı izleme yapılabilir.

journalctl -f: Sistemde anlık olarak oluşan logları incelemek için kullanılır.

```
(kali@kali)-[~]
$ journalctl -f

Jan 04 15:20:03 kali systemd[1]: user-109.slice: Consumed 10.140s CPU time.
Jan 04 15:21:47 kali systemd[1]: Started system_monitoring.service - System Monitoring Service.
Jan 04 15:21:48 kali systemd[1]: system_monitoring.service: Deactivated successfully.
Jan 04 15:23:47 kali systemd[1]: Started system_monitoring.service - System Monitoring Service.
Jan 04 15:23:48 kali systemd[1]: system_monitoring.service: Deactivated successfully.
Jan 04 15:25:01 kali CRON[12275]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
Jan 04 15:25:01 kali CRON[12279]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
Jan 04 15:25:02 kali CRON[12275]: pam_unix(cron:session): session closed for user root
Jan 04 15:25:12 kali systemd[1]: Started system_monitoring.service - System Monitoring Service.
Jan 04 15:25:14 kali systemd[1]: system_monitoring.service: Deactivated successfully.
Jan 04 15:26:47 kali systemd[1]: Started system_monitoring.service - System Monitoring Service.
Jan 04 15:26:48 kali systemd[1]: system_monitoring.service: Deactivated successfully.
Jan 04 15:27:00 kali dbus-daemon[1318]: [session uid=1000 pid=1318] Activating via systemd: service name='org.freedesktop.portal.Desktop' unit='xdg-desktop-portal.service' requested by ':1.62' (uid=1000 pid=13229 comm="/usr/lib/firefox-esr/firefox-esr")
Jan 04 15:27:01 kali systemd[1295]: Starting xdg-desktop-portal.service - Portal service ...
Jan 04 15:27:01 kali dbus-daemon[1318]: [session uid=1000 pid=1318] Activating via systemd: service name='org.freedesktop.portal.Documents' unit='xdg-document-portal.service' requested by ':1.63' (uid=1000 pid=13283 comm="/usr/libexec/xdg-desktop-portal")
Jan 04 15:27:01 kali systemd[1295]: Starting xdg-document-portal.service - flatpak document portal service ...
Jan 04 15:27:01 kali dbus-daemon[1318]: [session uid=1000 pid=1318] Activating via systemd: service name='org.freedesktop.impl.portal.PermissionStore' unit='xdg-permission-store.service' requested by ':1.64' (uid=1000 pid=13288 comm="/usr/libexec/xdg-document-portal")
Jan 04 15:27:01 kali systemd[1295]: Starting xdg-permission-store.service - sandboxed app permission store ...
Jan 04 15:27:01 kali dbus-daemon[1318]: [session uid=1000 pid=1318] Successfully activated service 'org.freedesktop.impl.portal.PermissionStore'
Jan 04 15:27:01 kali systemd[1295]: Started xdg-permission-store.service - sandboxed app permission store.
Jan 04 15:27:01 kali dbus-daemon[1318]: [session uid=1000 pid=1318] Successfully activated service 'org.freedesktop.portal.Documents'
```

`sudo` komutu, Linux sistemlerinde bir kullanıcıya yönetici **root yetkileriyle** komut çalıştırma izni verir. Bu, sistemi değiştirme, yazılım yükleme veya kritik ayarları değiştirme gibi işlemleri güvenli bir şekilde yapmayı sağlar. Yetkili kullanıcılar, **`sudo`** kullanarak yalnızca belirli komutlar için geçici olarak root yetkisi alabilir. Sudo yetkisine sahip kullanıcıları incelemek için **/etc/sudoers** dosyası incelenebilir. Fakat bu dosyayı incelemek için de root yetkilerine sahip olmak gereklidir. Aksi halde permission denied hatası alınır. Sudo yetkisi almış kullanıcılar sistemde tam yetkiye sahip olduğu için istedikleri işlemleri hemen halledebilir.

```
(kali@kali)-[~]
$ cat /etc/sudoers
cat: /etc/sudoers: Permission denied

(kali@kali)-[~]
$ sudo su
[sudo] password for kali:
(root@kali)-[/home/kali]
# cat /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
```

Linux sistemlerde kimlik doğrulaması yapmış her kullanıcı için bir log tutulur. Bu loglar authentication log olarak tutulur. Bu loglar **/var/log/auth.log** altına kaydedilir. Dosya boyutundan dolayı inceleme yapmak için head, tail, more, less gibi komutları kullanmak yardımcı olabilir. Sistem yapılandırmasına göre auth.log dosyası yerine journalctl de authentication loglarını tutabilir. **journalctl | grep authentication** komutuyla authentication logları incelenebilir. Özellikle bir servis için inceleme yapılmak isteniyorsa ssh gibi **-u** parametresiyle servis adı belirtilerek loglar incelenebilir.

```
(root@kali)-[/var/log/sysstat]
# journalctl | grep "authentication failure"

Sep 16 08:25:34 kali sudo[1751]: pam_unix(sudo:auth): authentication failure; logname=kali uid=1000 euid=0 tty=/dev/pts/0 ruser=kali rhost= user=kali
Sep 16 08:55:30 kali lightdm[11744]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
Oct 01 15:29:45 kali lightdm[1152]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:0 ruser= rhost= user=kali
Oct 08 08:00:48 kali lightdm[30901]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
Oct 08 08:01:03 kali sudo[31335]: pam_unix(sudo:auth): authentication failure; logname=kali uid=1000 euid=0 tty=/dev/pts/0 ruser=kali rhost= user=kali
Oct 09 06:23:57 kali sudo[48537]: pam_unix(sudo:auth): authentication failure; logname=kali uid=1000 euid=0 tty=/dev/pts/0 ruser=kali rhost= user=kali
Oct 13 06:50:57 kali lightdm[12931]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
Oct 26 09:34:14 kali lightdm[146260]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
Oct 26 09:34:31 kali lightdm[148077]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
Oct 27 04:03:28 kali sudo[3158]: pam_unix(sudo:auth): authentication failure; logname=kali uid=1000 euid=0 tty=/dev/pts/0 ruser=kali rhost= user=kali
Dec 26 03:34:37 kali lightdm[11615]: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:1 ruser= rhost= user=kali
```

Örneğin ilk log da sudo kullanıcısıyla bir işlem yapılmaya çalışılmış fakat kimlik doğrulama başarısız olmuştur. Altındaki lightdm logunda ise bir kullanıcının giriş yapmaya çalıştığını ancak kimlik doğrulamanın başarısız olduğunu gösteriyor. LightDM, sistemin giriş ekranını sağlayan bir oturum yöneticisidir. Bu loglar incelenerek sisteme yapılan oturum girişimleri tespit edilebilir.

Sistemin saat dilimini öğrenmek için windows'ta olduğu gibi bir timezone alanı bulunur. Bu alanı görmek için **/etc/timezone** dizinin cat ile okumak yeterli olacaktır.

Network interfaceleri hakkında bilgi sahibi olmak için **/etc/network/interfaces** altına bakılabilir. Farklı arayüzlerin mac ve ip adresleri bilgilerine ulaşmak için **ip address show** komutu kullanılabilir.

```
(root@kali)-[/etc/network]
# ip address show

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host proto kernel_lo
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:26:f2:32 brd ff:ff:ff:ff:ff:ff
    scope global dynamic noprefixroute eth0
    valid_lft 1335sec preferred_lft 1335sec
    inet6 [redacted] scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Ayrıca canlı bir sistemdeki aktif bağlantıları görmek için netstat komutu kullanılabilir. netstat -natp komutuyla sistemdeki bağlantıları görünür. -n, port numaralarını ve IP adreslerini numerik formatta gösterir; -a, tüm bağlantıları gösterir, dinleme (listen) bağlantıları da dahil olmak üzere; -t, TCP bağlantılarını gösterir; -p, bağlantıyı oluşturan sürecin adını/pid'sini gösterir.

Sistemde anlık olarak çalışan tüm processleri görmek için **ps -aux** komutu kullanılır. Sistemde çalışan processleri incelemek ve bilgi dışında olan processleri tespit etmek çok önemlidir. Tespit edilen processler pid'si kullanılarak durdurulabilir.

```
(kali㉿kali)-[~]
$ ps aux
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root            1  0.1  0.3 22944 13008 ?        Ss   06:45   0:07 /sbin/init splash
root            2  0.0  0.0      0      0 ?        S    06:45   0:00 [kthreadd]
root            3  0.0  0.0      0      0 ?        S    06:45   0:00 [pool_workqueue_release]
root            4  0.0  0.0      0      0 ?        I<   06:45   0:00 [kworker/R-rcu_g]
root            5  0.0  0.0      0      0 ?        I<   06:45   0:00 [kworker/R-rcu_p]
root            6  0.0  0.0      0      0 ?        I<   06:45   0:00 [kworker/R-slub_]
root            7  0.0  0.0      0      0 ?        I<   06:45   0:00 [kworker/R-netns]
root           11  0.0  0.0      0      0 ?        I    06:45   0:00 [kworker/u64:0-ext4-rsv-conversion]
root           12  0.0  0.0      0      0 ?        I<   06:45   0:00 [kworker/R-mm_pe]
root           13  0.0  0.0      0      0 ?        I    06:45   0:00 [rcu_tasks_kthread]
root           14  0.0  0.0      0      0 ?        I    06:45   0:00 [rcu_tasks_rude_kthread]
root           15  0.0  0.0      0      0 ?        I    06:45   0:00 [rcu_tasks_trace_kthread]
root           16  0.0  0.0      0      0 ?        S    06:45   0:00 [ksoftirqd/0]
root           17  0.0  0.0      0      0 ?        I    06:45   0:04 [rcu_preempt]
root           18  0.0  0.0      0      0 ?        S    06:45   0:00 [migration/0]
root           19  0.0  0.0      0      0 ?        S    06:45   0:00 [idle_inject/0]
root           20  0.0  0.0      0      0 ?        S    06:45   0:00 [cpuhp/0]
root           21  0.0  0.0      0      0 ?        S    06:45   0:00 [cpuhp/1]
root           22  0.0  0.0      0      0 ?        S    06:45   0:00 [idle_inject/1]
root           23  0.0  0.0      0      0 ?        S    06:45   0:01 [migration/1]
root           24  0.0  0.0      0      0 ?        S    06:45   0:00 [ksoftirqd/1]
root           27  0.0  0.0      0      0 ?        S    06:45   0:00 [cpuhp/2]
root           28  0.0  0.0      0      0 ?        S    06:45   0:00 [idle_inject/2]
root           29  0.0  0.0      0      0 ?        S    06:45   0:01 [migration/2]
root           30  0.0  0.0      0      0 ?        S    06:45   0:00 [ksoftirqd/2]
```

Çıktıdaki her bir sütün farklı bir anlama gelir.

USER: İşlemi başlatan kullanıcı adı

PID: İşlem Kimliği (Process ID), yani işlemin benzersiz numarası

%CPU: CPU kullanım yüzdesi

%MEM: Bellek kullanım yüzdesi

VSZ: Toplam sanal bellek boyutu (kilobyte cinsinden)

RSS: Resident Set Size, yani işlemin fiziksel bellek boyutu (kilobyte cinsinden)

TTY: İşlemin bağlı olduğu terminal

STAT: İşlem durumu (örneğin, çalışıyor, beklemede, durmuş)

START: İşlemin başlatılma zamanı

TIME: İşlem tarafından kullanılan toplam CPU zamanı

COMMAND: İşlemi başlatan komut veya program

CPU ve bellek kullanım yüzdelerini incelemek anomali tespitinde büyük önem taşır. Ayrıca sistemde çalıştırılan komutları da incelemek gereklidir. Anormal bir komut çalıştıran programlar tespit edilip aksiyon alınabilir.

Linux sistemde kullanıcıların terminalde girdiği komutlar sistem kabuğuna göre bir history dosyasına kaydedilir. Örneğin kali Linux eski versiyonları varsayılan olarak bash kabuğunu kullanırken yeni versiyonlar zsh kabuğunu kullanır. **.zsh_history** dosyası incelendiğinde terminalde çalıştırılan komutlar incelenip anomali tespiti yapılır.

```
(root@kali)-[~]
# cat .zsh_history
nano /etc/resolv.conf
ping google.com
sudo systemctl restart NetworkManager\
ping google.com
nano /etc/resolv.conf
cd /etc
ls
nano /etc/hosts
cd
cd /home/kali/Desktop
zip -r shell.zip
zip -r shell.zip reversephp
zip -r shell.zip reverse.php
setxkbmap tr
cd /etc
ls
cat crontab
nano crontab
systemctl restart cron.service
```

Şekilde networkmanager'ı yeniden başlattıktan sonra bir Shell dosyası zipten çıkarılmış ve daha sonra crontab servisi yeniden başlatılmıştır.

Sistemde syslog servisi aktifse bu loglar incelenebilir. Syslog, sistem başlangıç mesajları, kernel aktiviteleri, ağ bağlantıları ve sistem hataları gibi geniş bir yelpazede olayları kaydeden kapsamlı bir günlük dosyasıdır. Syslog dosyasını incelemek, sistem olaylarını izlemek, belirli hizmetlerin ne zaman başlatıldığını veya durdurulduğunu tespit etmek ve yetkisiz girişler veya sistem hataları gibi olağandışı davranışları belirlemek için kritik bir öneme sahiptir. Eğer sistemde syslog aktif değilse journalctl üzerinden inceleme yapılabilir.

Cron jobs, belirli aralıklarla otomatik olarak çalışan zamanlanmış görevlerdir. Bu görevler genellikle sistem yöneticileri tarafından bakım, yedekleme ve diğer otomatik işlemler için yapılandırılır. Cron jobları analiz etmek, belirli aralıklarla çalışacak şekilde ayarlanmış yetkisiz veya kötü amaçlı görevlerin varlığını ortaya çıkarabilir. **/var/spool/cron/crontabs/** dizininde sistemde var olan cron joblar incelenir. **Crontab -l** komutuyla da o anki kullanıcı için oluşturulmuş cronjoblar listelenir.

```
(root@kali)-[/var/log]
# crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
# m h dom mon dow command
* * * * * ping google.com >> /var/log/google_ping.log 2>&1
```

Bu sistemde her dakikada bir google'a ping atacak ve loglarını /var/log/Google_ping.log adlı bir dosyaya kaydedecek bir crontab oluşturulmuş. İncelenilen sistemde herhangi bir cronjob yoksa no crontab for user uyarısı verir. Bu sistemde bu kullanıcı için herhangi bir zamanlanmış görev olmadığı gösterir.

SSH (Secure Shell), güvenli uzaktan giriş ve komut yürütme için kullanılan bir protokoldür. SSH ile ilişkili loglar ve yapılandırma dosyaları, uzaktan erişim girişimlerini, zayıf veya ele geçirilmiş anahtarların kullanımını ve SSH hizmetinin güvenlik durumunu anlamak için kritik öneme sahiptir. Bu incelemeler, saldırganların SSH üzerinden bir sisteme kalıcı erişim sağlamış olabilecekleri durumları tespit etmek için önemlidir. /etc/ssh dizini altında ssh sunucu yapılandırma dosyaları bulunur. /etc/ssh/sshd_config; SSH bağlantıları, şifreleme yöntemleri, portlar, kimlik doğrulama seçenekleri gibi ayarlar yapılan SSH sunucusunun yapılandırma dosyasıdır. /etc/ssh/ssh_config; SSH istemcilerinin sunucularla nasıl bağlantı kuracağına dair ayarları içeren SSH istemci yapılandırma dosyasıdır ve genellikle istemci tarafında bulunur.

Ssh üzerinden giriş denemesi loglarını ise auth.log dosyasından incelenir. **sudo cat /var/log/auth.log | grep sshd** komutu auth.log dosyasını yazdırırken sadece sshd loglarını görmek için grep ile bir filtreleme yapar.

Paket yönetimi loglarını analiz etmek, yetkisiz yazılımların kurulumunu tespit etmeye, olası zararlı yazılımlar veya backdoorları belirlemeye ve sistemdeki değişikliklerin zaman içindeki izini sürmeye yardımcı olur. Çünkü saldırgan sistemi ele geçirmek ve kontrol altında tutmak amacıyla kötü amaçlı paketler yüklemiş olabilir. Bu logları incelemek için sistemin paket yönetim aracını bilmek gerekir. Debian tabanlı sistemlerde bu dpkg'dir ya da red hat tabanlı sistemlerde yum'dur.

Paket yönetim logları debian sistemlerde /var/log/dpkg.log olarak tutulur. Buradan sisteme kurulan paketler hakkında incelemeler yapıp anomali tespiti yapılabilir.

```
(root@kali)-[/var/log]
# cat dpkg.log
2025-01-05 14:33:19 startup archives unpack
2025-01-05 14:33:25 install libestr0:amd64 <none> 0.1.11-1+b2
2025-01-05 14:33:25 status triggers-pending libc-bin:amd64 2.38-10
2025-01-05 14:33:25 status half-installed libestr0:amd64 0.1.11-1+b2
2025-01-05 14:33:26 status unpacked libestr0:amd64 0.1.11-1+b2
2025-01-05 14:33:26 install libfastjson4:amd64 <none> 1.2304.0-2
2025-01-05 14:33:26 status half-installed libfastjson4:amd64 1.2304.0-2
2025-01-05 14:33:26 status unpacked libfastjson4:amd64 1.2304.0-2
2025-01-05 14:33:26 install liblognorm5:amd64 <none> 2.0.6-4+b2
2025-01-05 14:33:26 status half-installed liblognorm5:amd64 2.0.6-4+b2
2025-01-05 14:33:26 status unpacked liblognorm5:amd64 2.0.6-4+b2
2025-01-05 14:33:26 install rsyslog:amd64 <none> 8.2410.0-1
2025-01-05 14:33:26 status half-installed rsyslog:amd64 8.2410.0-1
2025-01-05 14:33:27 status triggers-pending kali-menu:all 2023.4.7
2025-01-05 14:33:27 status triggers-pending man-db:amd64 2.12.1-1
2025-01-05 14:33:27 status unpacked rsyslog:amd64 8.2410.0-1
2025-01-05 14:33:27 startup packages configure
2025-01-05 14:33:27 configure libestr0:amd64 0.1.11-1+b2 <none>
2025-01-05 14:33:27 status unpacked libestr0:amd64 0.1.11-1+b2
2025-01-05 14:33:27 status half-configured libestr0:amd64 0.1.11-1+b2
2025-01-05 14:33:27 status installed libestr0:amd64 0.1.11-1+b2
2025-01-05 14:33:27 configure libfastjson4:amd64 1.2304.0-2 <none>
2025-01-05 14:33:27 status unpacked libfastjson4:amd64 1.2304.0-2
2025-01-05 14:33:27 status half-configured libfastjson4:amd64 1.2304.0-2
2025-01-05 14:33:27 status installed libfastjson4:amd64 1.2304.0-2
2025-01-05 14:33:27 configure liblognorm5:amd64 2.0.6-4+b2 <none>
2025-01-05 14:33:27 status unpacked liblognorm5:amd64 2.0.6-4+b2
```

Üstteki fotoğrafa bakıldığında libestro, libfastjson4, rsyslog, liblognorm5 gibi paketlerin yüklendiği ve yapılandırıldığı görülüyor. Tarih ve zaman damgaları ise yanlarında görülmektedir. Analiz sırasında belirli bir tarih aralığında filtreleme yapmak için kullanılabilir.

HAZIRLAYAN

Ayşe BALCI

[Linkedin](#)