

Ahtapot

Saldırı Tespit ve Önleme Sistemi

Kurulum ve Kullanımı

Saldırı tespit sistemi; Suricata, Puledpork, Barnyard2, Mysql ve Snorby yazılım bileşenlerinden oluşmaktadır.

Suricata: Açık kaynak saldırı tespit ve önleme sistemidir.

Puledpork: Açık kaynak Suricata ve Snort imza güncellemelerini ve yönetimini sağlayan araçtır.

Barnyard2: Suricata'nın unified2 formatında ürettiği uyarı kayıtlarını mysql veritabanına yazma işlemini yapar. Aynı zamanda imza bilgilerini de veri tabanına ekler.

Snorby: Barnyard2'nin veri tabanına yazdığı saldırı uyarılarının web tabanlı olarak görüntülenmesini/izlenmesini sağlar.

1- Suricata Kurulumu ve Kullanımı

Ayar dosyası içerisinde aşağıdaki değişkenler ayarlanır. Ayar dosyasındaki yapılandırmaya göre ids yada ips olarak çalıştırılabilir.

IDS Modunun aktif edilmesi için **suricata_mode: ids** olarak ayarlanır.

IDS modunda suricata'nın dinleyeceği her bir interface **suricata_pcap_interfaces:** bölümü altında ayrı ayrı tanımlanır.

IPS modunda ise pcap interface tanımlanırının yapılmasına gerekli değildir.

IPS modunda kullanabilmek için **suricata_mode: ips** olarak ayarlanır.

Yapılandırma ayarları bittikten sonra ansible-playbook çalıştırılır.

IDS Yapılandırması

Ayar Dosyası: /etc/ansible/roles/suricata/vars/main.yml

Parametreler:

suricata_mode: ids #ids|ips

suricata_pcap_interfaces:

- int: enp0s3

- buffer_size: 16777216

- bpf_filter: "tcp and port 25"

- checksum_checks: auto

- threads: 16

- promisc: no

- snaplen: 1518

- int: enp0s8

- buffer_size: 16777216

- bpf_filter: "tcp and port 25"

- checksum_checks: auto

threads: 16
promisc: no
snaplen: 1518

Komut: ansible-playbook /etc/ansible/playbooks/suricata.yml

IPS yapılandırması

IPS için Suricata'nın NFQUEUE modu kullanılır. NFQ iptables'in bir özelliğidir. Bu özellik sayesinde trafik Queue'lara yönlendirilir. Suricata bu Queue'da trafiği **inline** analiz eder ve aktif olan imzalarda **drop** olanları düşürür, **alert** olanlar için uyarı üretir ve trafiği iptables'a geri verir.

Birden fazla Queue suricata daemon'na yönlendirilebilir. Queue sayısı CPU core sayısına eş olacak veya bir eksiği olacak şekilde ayarlanması önerilir.

```
NPROC=$(/usr/bin/nproc)  
NPROC=$((NPROC - 1)) && echo $NPROC
```

NFQUEUE iptables kurallarının yazımı **FWBuilder** üzerinden yapılır. **NAT/ROUTER** veya **Bridge** mod olarak konumlanmış güvenlik duvarlarında farklı kurallar yazılır.

Firewall kuralları ayarlandıktan sonra Suricata'nın NFQUEUE modunda çalışması için servis ayarları yapılır.

Ayar Dosyası: /etc/ansible/roles/suricata/vars/main.yml

Parametreler:

suricata_mode: ips #ids|ips

Komut: ansible-playbook /etc/ansible/playbooks/suricata.yml

FWBuilder üzerinde aşağıdaki yol takip edilerek kurallar eklenir ve politikalar ilgili sistemlere gönderilir.

NAT/ROUTER Mod IPS

- 1- Firewalls > Firewall_Obje > Policy > Çift Klik > Top Rule Set [] işareti kaldırılır.
- 2- Firewalls > Firewall_Obje > Sağ Klik > New Policy Rule Set denilerek yeni policy oluşturulur.
Name: 00_SuricataNFQ
[x] Top Rule Set işaretlenir.
- 3- Eklenen yeni Policy tablosuna 1 kural eklenir ve **Options** kısmında **logging off** seçilir.
- 4- Eklenen kuralın **Rule Options** kısmında aşağıdaki işlemler yapılır.
a- Assume firewall is part of "any" for this rule only: **off**
b- Stateless Rule [] işareti kaldırılır.
- 5- Kuralın ACTION kısmında Custom seçilir ve aşağıdaki değer yazılır.

#nproc komut ile cpu core sayısı belirlenir ve aşağıdaki kuraldaki uygun alana yazılır.

/usr/bin/nproc

Cpu Core Sayısı: 4

```
-j NFQUEUE --queue-balance 1:4 --queue-bypass
```

#Eğer 1 tane cpu core var ise kural'ın ACTION kısmı aşağıdaki gibi yazılır.

```
-j NFQUEUE --queue-num 1 --queue-bypass
```

Not: **queue-bypass** parametresi ile suricata servisi çalışmıyor ise trafiği alttaki kurallardan işletmeye devam eder.

[Suricata NFQUEUE Yapılandırması](#)

Bridge Mod IPS

Bridge mod IPS ayarları için öncelikle Bridge interface aşağıdaki yapılandırma örneğindeki gibi yapılandırılır.

Not: Bridge yapılandırması için bridge-utils ve ethtool paketlerinin sistemde kurulu olması gerekmektedir.

Hangi iki fiziksel interface bridge interface'in üyesi olacak ise bridge_ports parametresi ile tanımlanır. Ethtool ile özel ayarlar yapılması istenen fiziksel interface'ler tanımlanır.

Ayar Dosyası: /etc/network/interfaces

```
auto br0
iface br0 inet manual
    bridge_ports eth1 eth2
    bridge_stp on
    bridge_fd 5

#Advanced Interface Settings
#post-up /sbin/ethtool -K eth1 tx off rx off sg off gso off gro off 1>/dev/null
#post-up /sbin/ethtool -K eth2 tx off rx off sg off gso off gro off 1>/dev/null

#SYSCTL Ayarları yapılır.
post-up /sbin/sysctl -w net.bridge.bridge-nf-call-arptables=1 &> /dev/null
post-up /sbin/sysctl -w net.bridge.bridge-nf-call-ip6tables=1 &> /dev/null
post-up /sbin/sysctl -w net.bridge.bridge-nf-call-iptables=1 &> /dev/null
post-up /sbin/sysctl -w net.bridge.bridge-nf-filter-pppoe-tagged=1 &> /dev/null
post-up /sbin/sysctl -w net.bridge.bridge-nf-filter-vlan-tagged=1 &> /dev/null
post-up /sbin/sysctl -w net.bridge.bridge-nf-pass-vlan-input-dev=1 &> /dev/null
```

Ayarlar kayıt edilir ve bridge interface UP edilir.

ifup br0

IPTables'in Bridge olarak çalışabilmesi için modüllerin aktif edilmesi gerekir.

```
if [ -f /etc/modprobe.conf ]; then
    /bin/sed -i '/^ip_tables/h;${x;/^${s//ip_tables/;H};x}' /etc/modprobe.conf
    /bin/sed -i '/^x_tables/h;${x;/^${s//x_tables/;H};x}' /etc/modprobe.conf
    /bin/sed -i '/^br_netfilter/h;${x;/^${s//br_netfilter/;H};x}' /etc/modprobe.conf
    /bin/sed -i '/^xt_physdev/h;${x;/^${s//xt_physdev/;H};x}' /etc/modprobe.conf

    /sbin/modprobe ip_tables
    /sbin/modprobe x_tables
    /sbin/modprobe br_netfilter
    /sbin/modprobe xt_physdev
else
    echo "ip_tables" >> /etc/modprobe.conf
```

```
echo "x_tables" >> /etc/modprobe.conf
echo "br_netfilter" >> /etc/modprobe.conf
echo "xt_physdev" >> /etc/modprobe.conf

/sbin/modprobe ip_tables
/sbin/modprobe x_tables
/sbin/modprobe br_netfilter
/sbin/modprobe xt_physdev
fi
```

Bridge interface yapılandırması yapıldıktan sonra ve iptables modülleri aktif edildikten sonra FWBuilder ile kurallar yazılır.

- 1- Firewalls > Firewall_Obje > Policy > Çift Klik > Top Rule Set [] işareti kaldırılır.
- 2- Firewalls > Firewall_Obje > Sağ Klik > New Policy Rule Set denilerek yeni policy oluşturulur.
Name: 00_SuricataNFQ
[x] Top Rule Set işaretlenir.
- 3- Eklenen yeni Policy tablosuna 2 kural eklenir ve **Options** kısmında **logging off** seçilir.
- 4- Eklenen kuralların **Rule Options** kısmında aşağıdaki işlemler yapılır.
a- Assume firewall is part of "any" for this rule only: **off**
b- Stateless Rule [] işareti kaldırılır.
- 5- Eklenen kuralların **Direction** kısımları düzenlenir.
a- 1 nolu kuralın Direction'u **OUTBOUND** olarak ayarlanır.
b- 2 nolu kuralın Direction'u **INBOUND** olarak ayarlanır.
- 6- Kuralların ACTION kısmında Custom seçilir ve aşağıdaki değer yazılır.
#nproc komut ile cpu core sayısı belirlenir ve aşağıdaki kuraldaki uygun alana yazılır.

/usr/bin/nproc

Cpu Core Sayısı: 4

a- OUTBOUND ACTION

-m physdev --physdev-in eth1 --physdev-out eth2 -j NFQUEUE --queue-bypass --queue-balance 1:4

b- INBOUND ACTION

-m physdev --physdev-in eth2 --physdev-out eth1 -j NFQUEUE --queue-bypass --queue-balance 1:4

#Eğer 1 tane cpu core var ise kural'ın ACTION kısmı aşağıdaki gibi yazılır.

a- OUTBOUND ACTION

-m physdev --physdev-in eth1 --physdev-out eth2 -j NFQUEUE --queue-bypass --queue-num 1

b- INBOUND ACTION

-m physdev --physdev-in eth2 --physdev-out eth1 -j NFQUEUE --queue-bypass --queue-num 1

Not: queue-bypass parametresi ile suricata servisi çalışmıyor ise trafiği alttaki kurallardan işletmeye devam eder.

Suricata NFQUEUE Yapılandırması

Suricata servis dosyası düzenlenir ve Queue Sayısı kadar -q parametresi verilerek düzenleme yapılır.

Suricata Servis Dosyası: /lib/systemd/system/suricata.service

Bu dosyanın içerisindeki ExecStart bölümünde yer alan **--af-packet** parametresi silinir. Bunun yerine Queue sayısı kadar -q parametresi tekrarlanır.

Önceki

ExecStart=/usr/bin/suricata -D **--af-packet** -c /etc/suricata/suricata.yaml --pidfile
/var/run/suricata.pid

Sonraki

ExecStart=/usr/bin/suricata -D -q 1 -q 2 -q 3 -q 4 -c /etc/suricata/suricata.yaml --pidfile
/var/run/suricata.pid

Ayarlar kayıt edildikten sonra servis yeniden başlatılır.

systemctl daemon-reload

systemctl restart suricata

Ayar dosyası: /etc/suricata/suricata.yaml

Suricata'nın bütün ayarlarının yer aldığı dosyadır.

İmza Dizini: /etc/suricata/rules

Bütün imzalar bu dizin altında yer alır.

Kayıt Dizini: /var/log/suricata

Fast Log: /var/log/suricata/fast.log

Bu dosyada anlık kısa uyarı kayıtları görüntülenir.

Eve Log: /var/log/suricata/eve.json

Bu dosyada anlık detaylı json formatında uyarı kayıtları görüntülenir.

Stats log: /var/log/suricata/stats.log

Bu dosyada Suricata servisi ve bazı istatistikleri ile ilgili bilgiler görüntülenir.

2- Pulledpork Kurulum ve Kullanımı

Pulledpork kurulmadan önce suricata kurulmuş olmalıdır. Ayar dosyasındaki alanlar düzenlenir ve ansible playbook çalıştırılır.

pulledpork_etpro_key: Var ise emergingthreats lisans anahtarı bu alana girilir. Lisans yok ise **open** yazılır.

pulledpork_rule_url: İmzaların bulunduğu URL adresi yazılır.

pulledpork_rule_file: Tanımlanan url altındaki imza dosyasını adı.

pulledpork_exclude_rules: Hariç bırakılacak imza kategorileri.

- pass

pulledpork_update_frequency: İmza güncelleme sıklığı. (Minutely, hourly, daily, monthly, weekly, yearly)

Ayar Dosyası: /etc/ansible/roles/pulledpork/vars/main.yml

Parametreler:

pulledpork_etpro_key: open

pulledpork_rule_url: https://rules.emergingthreats.net/

pulledpork_rule_file: emerging.rules.tar.gz

pulledpork_exclude_rules:

- pass

pulledpork_update_frequency: daily

Komut: ansible-playbook /etc/ansible/playbooks/pulledpork.yml

Pulledpork Ayar Dizini: /etc/pulledpork/

pulledpork.conf: Pulledpork'un genel ayarlarının bulunduğu dosyadır.

disablesid.conf: Disable edilecek imzaların SID'leri bu dosyaya yazılır.

dropsid.conf: Drop edilmesi istenilen imzaların SID'leri bu dosyaya yazılır.

enablesid.conf: Aktif edilmek istenen imzaların SID'leri bu dosyaya yazılır.

Pulledpork Kayıt Dizini: /var/log/pulledpork/

Backup Dizini: /var/lib/pulledpork-backup/

Pulledpork Perl Script: /usr/local/sbin/pulledpork.pl

Pulledpork güncelleme script: /usr/local/sbin/pulledpork-rule-updater.sh

Periyodik imza güncellemeleri için bu script çağrılır. Bu script pulledpork.pl script'ini çağırır ve sonrasında yapılması gereken servislerin yeniden başlatılması gibi diğer işlemleri yapar.

Pulledpork Servisi: /lib/systemd/system/pulledpork.service

Bu servis sadece tek bir sefer çalıştırılmak üzere yapılandırılmıştır. Zamanlanmış güncellemeler için kullanılır.

Pulledpork Zamanlanmış Görev: /lib/systemd/system/pulledpork.timer

Pulledpork servisini belirlenen zamanlarda çalıştırarak imzaların güncellenmesini sağlar.

3- Snorby Kurulum ve Kullanımı

Snorby kurulmadan önce mysql server, ansible kullanılarak kurulur. Ayar dosyasındaki alanlar düzenlenir ve ansible playbook çalıştırılır.

snorby_company_name: Kurum veya Organizasyon adı girilir.

snorby_domain: Alan adı girilir. Örn. example.com

snorby_email: Yönetici kullanıcının e-posta adresi girilir. Bu aynı zamanda kullanıcı adı olarak kullanılır.

snorby_password: Yönetici kullanıcının parolası belirlenir.

snorby_mysql_user: Snorby yazılımının **snorby** database'ine erişim izni olan **mysql** kullanıcı adı girilir.

snorby_mysql_password: Mysql parolası girilir.

snorby_mysql_host: Mysql sunucusunun çalıştığı sunucunun IP adresi girilir.

Ayar Dosyası: /etc/ansible/roles/snorby/vars/main.yml

Parametreler:

snorby_company_name: Ahtapot

snorby_domain: ahtapot.example.com

snorby_email: snorby@example.com

snorby_password: snorby

snorby_mysql_user: snorby

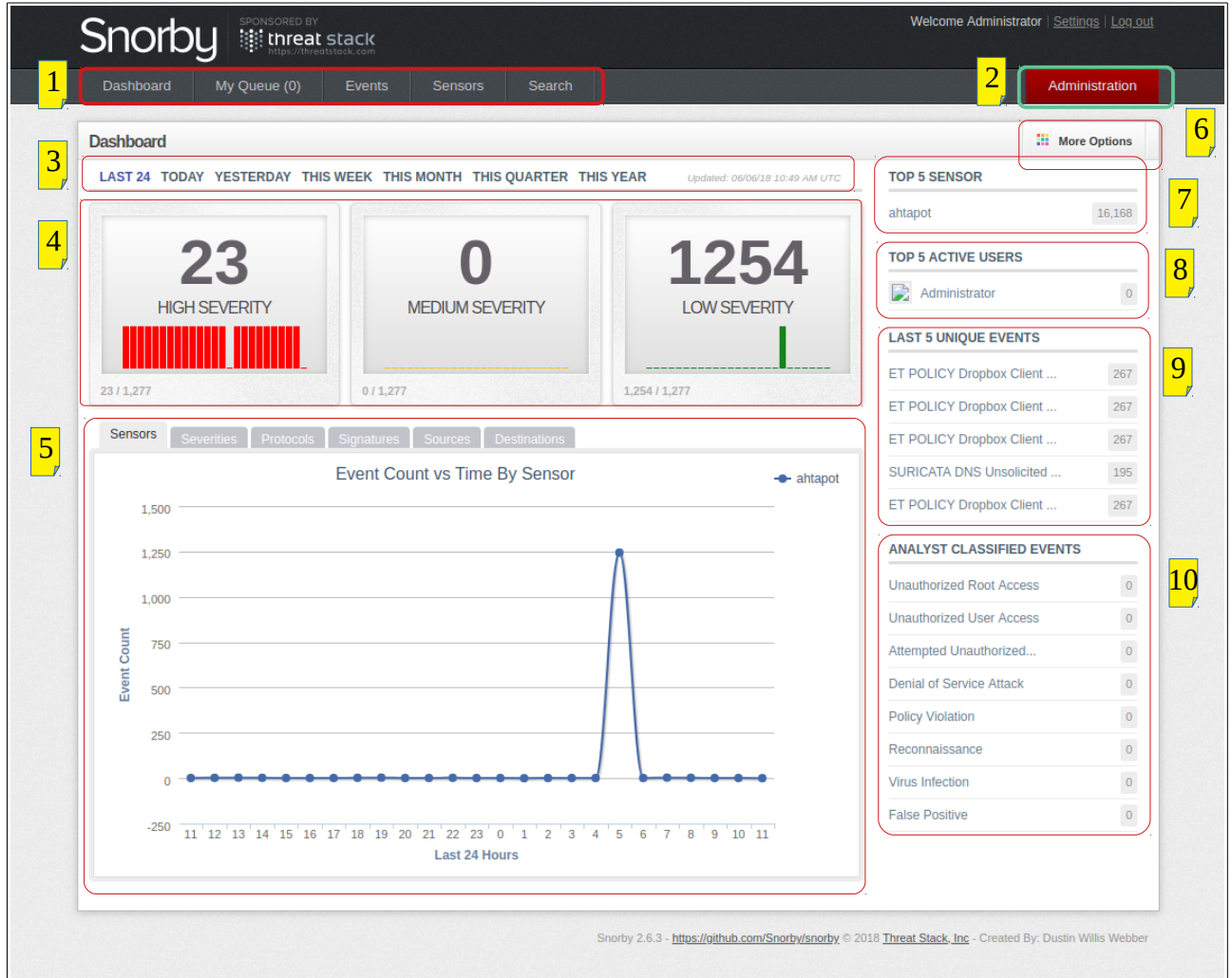
snorby_mysql_password: snorby

snorby_mysql_host: localhost

Komut: ansible-playbook /etc/ansible/playbooks/snorby.yml

Kurulum tamamlandıktan sonra snorby'ye web erişim adresi kullanılarak bağlanılır.

Snorby Web Adresi: http://cihaz_ip_adresi:3000



Dashboard

- 1- Snorby menü opsiyonlarının bulunduğu alandır. Bu alanda yer alan menüleri kullanarak uyarı ve sensör listestelerine ulaşılabilir. Ayrıca Search kısmı kullanılarak detaylı arama yapılabilir.
- 2- Snorby'nin genel ayarları, imza sınıfları ve servis durumları gibi ayarların yapılabildiği alt menülere erişim sağlar.
- 3- Dashboard üzerinde görüntülenen verinin tarih filtresidir.
- 4- Uyarıların kiritiklik seviyesine göre istatistiklerinin gösterildiği alandır. Üzerine tıklanarak detaylarına ulaşılabilir.
- 5- Pasta ve zaman çizelgesi tiplerine göre bazı istatistiklerin yer aldığı alandır.
- 6- PDF dışa aktarma, zaman filtresi gibi opsiyonların bulunduğu alandır.
- 7- Sistemde yer alan ve en çok uyarı kaydı olan 5 sesör'ün listelendiği alandır.
- 8- Snorby sistemini en çok kullanan 5 kullanıcının listelendiği alandır.
- 9- En son kaydedilen farklı türdeki 5 uyarı kaydının listelendiği alandır.
- 10- Uyarıların sınıflarına göre kategorize edilerek listelendiği alandır.

Events

Bütün saldırı uyarılarının görüntülendiği bölümdür.

Snorby

SPONSORED BY
threat stack
<https://threatstack.com>

Welcome Administrator | [Settings](#) | [Log out](#)

Dashboard

My Queue (0)

Events

Sensors

Search

Administration

Listing Sessions (20 unique unclassified sessions)

Hotkeys

Classify Event(s)

Filter Options

<input type="checkbox"/>	Sev.	Sensor	Source IP	Destination IP	Event Signature	Timestamp	Sessions
<input type="checkbox"/>	★ 1	ahtapot	192.168.3.123	192.168.3.255	ET POLICY Dropbox Client Broadcasting	12:40 PM	269
<input type="checkbox"/>	★ 3	ahtapot	8.8.8.8	192.168.3.61	SURICATA DNS Unsolicited response	11:46 AM	196
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.88.161	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	5:03 AM	4,955
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.92.152	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	5:03 AM	2,730
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.91.26	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	06/05/2018	276
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.88.149	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	06/04/2018	4,203
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.88.152	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	06/04/2018	1,042
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.91.23	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	06/03/2018	138
<input type="checkbox"/>	★ 3	ahtapot	192.168.3.61	91.189.88.162	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	06/01/2018	2,288
<input type="checkbox"/>	★ 2	ahtapot	81.21.167.2	192.168.3.61	Snort Alert [1:9000015:1]	05/30/2018	6
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	81.21.167.2	Snort Alert [1:9000014:1]	05/30/2018	39
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	157.240.20.35	Snort Alert [1:9000013:1]	05/30/2018	1
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	81.21.167.2	Snort Alert [1:9000013:1]	05/30/2018	8
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	8.8.8.8	DNS Query Detection: google.com	05/30/2018	14
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	35.234.101.161	ET POLICY curl User-Agent Outbound	05/30/2018	1
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	35.234.101.161	Snort Alert [1:9000011:1]	05/30/2018	1
<input type="checkbox"/>	★ 2	ahtapot	192.168.3.61	8.8.8.8	DNS Query Detection: mynet.com	05/30/2018	2
<input type="checkbox"/>	★ 1	ahtapot	192.168.3.123	224.0.0.252	ET P2P ThunderNetwork UDP Traffic	05/25/2018	1
<input type="checkbox"/>	★ 3	ahtapot	163.172.54.187	192.168.3.61	Filename Matched	05/25/2018	1
<input type="checkbox"/>	★ 3	ahtapot	35.234.101.161	192.168.3.61	Content Detection(http): Otomobil	05/25/2018	2

Uyarı Detayı

İlgili saldırı üzerine tıklanarak saldırı hakkında detay bilgiye ulaşılabilir.

Listing Sessions (20 unique unclassified sessions)

Hotkeys Classify Event(s) Filter Options

Sev.	Sensor	Source IP	Destination IP	Event Signature	Timestamp	Sessions
1	ahtapot	192.168.3.123	192.168.3.255	ET POLICY Dropbox Client Broadcasting	12:40 PM	269
3	ahtapot	8.8.8.8	192.168.3.61	SURICATA DNS Unsolicited response	11:46 AM	196
3	ahtapot	192.168.3.61	91.189.88.161	ET POLICY GNU/Linux APT User-Agent Outbound likely relate...	5:03 AM	4,955

IP Header Information

1 View All Sessions Perform Mass Classification Event Export Options Permalink

Source	Destination	Ver	Hlen	Tos	Len	ID	Flags	Off	TTL	Proto	Csum
192.168.3.61	91.189.88.161	4	5	0	173	0	0	0	0	6	16904

Signature Information

3

Generator ID	Sig. ID	Sig. Revision	Activity (15630/16171)	Category	Sig Info
1	2013504	5	96.65%	not-suspicious	Query Signature Database View Rule

TCP Header Information

Src Port	Dst Port	Seq	Ack	Off	Res	Flags	Win	Csum	URP
35465	80	0	0	5	0	0	0	8112	0

References

4

Type	Value
url	help.ubuntu.com/community/AptGet/Howto

Payload

5

Hex Ascii

```
00000000: 52 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 74 72 2e 61 72 63 68 69 R.HTTP/1.1..Host:.tr.archi
00000010: 76 65 2e 75 62 75 6e 74 75 2e 63 6f 6d 0d 0a 43 61 63 68 65 2d 43 6f 6e 74 72 ve.ubuntu.com..Cache-Contr
00000020: 6f 6c 3a 20 6d 61 78 2d 61 67 65 3d 30 0d 0a 41 63 63 65 70 74 3a 20 74 65 78 ol:.max-age=0..Accept:.tex
00000030: 74 2f 2a 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20 44 65 62 69 61 6e 20 41 50 t/*..User-Agent:.Debian.AP
00000040: 54 2d 48 54 54 50 2f 31 2e 33 20 28 31 2e 30 2e 31 75 62 75 6e 74 75 32 29 0d T-HTTP/1.3.(1.0.1ubuntu2).
00000050: 0a 0d 0a ...
```

Notes

6

Administrator Wednesday, Jun 06, 2018 at 1:34:44 PM UTC

Trafiği incelenmeli

Add A Note To This Event

1- Bu uyarı ile ilgili yapılabilecek işlemlerin opsiyonlarının bulunduğu alandır.

2- IP başlık bilgilerinin yer aldığı alandır. IP Adresi üzerine tıklayarak iIP adresine isim verebilir, bu IP adresi ile ilgili whois sorgusu gerçekleştirilebilir veya uyarı veri tabanındaki geçmiş olayları aranabilir.

3- İmzaya ilişkin bilgilerin bulunduğu alandır. Bu alandaki Query signature Database butonu ile imzanın referans URL adresine erişilebilir detayına bakılabilir. Ayrıca View Rule opsiyonu ile imzanın içeriği görüntülenebilir.

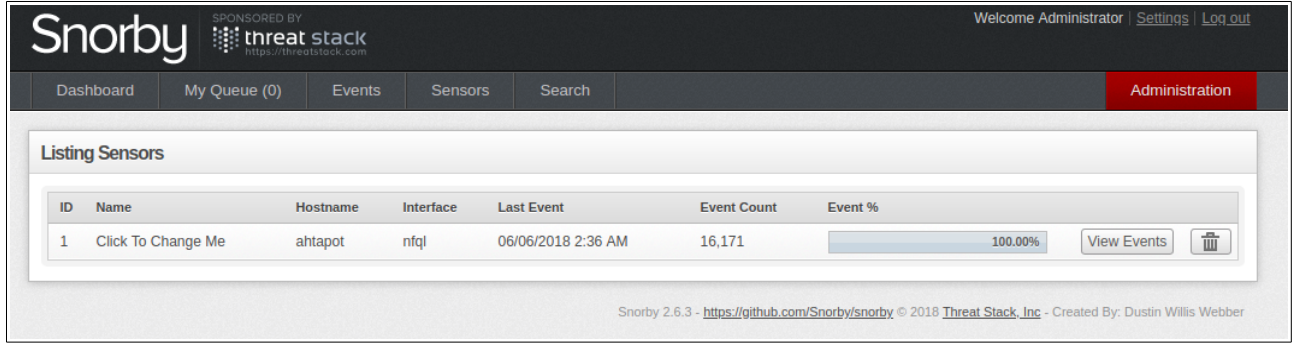
4- Referans Url adresinin bulunduğu bölümdür.

5- Uyarıya ilişkin trafiğin içeriğinin görüntülendiği alandır. Bu alanda ki içerik HEX formatında görüntülenebileceği gibi ASCII formatında da görüntülenebilir.

6- Uyarıya ilişkin yönetici notlarının eklendiği alandır.

Sensors

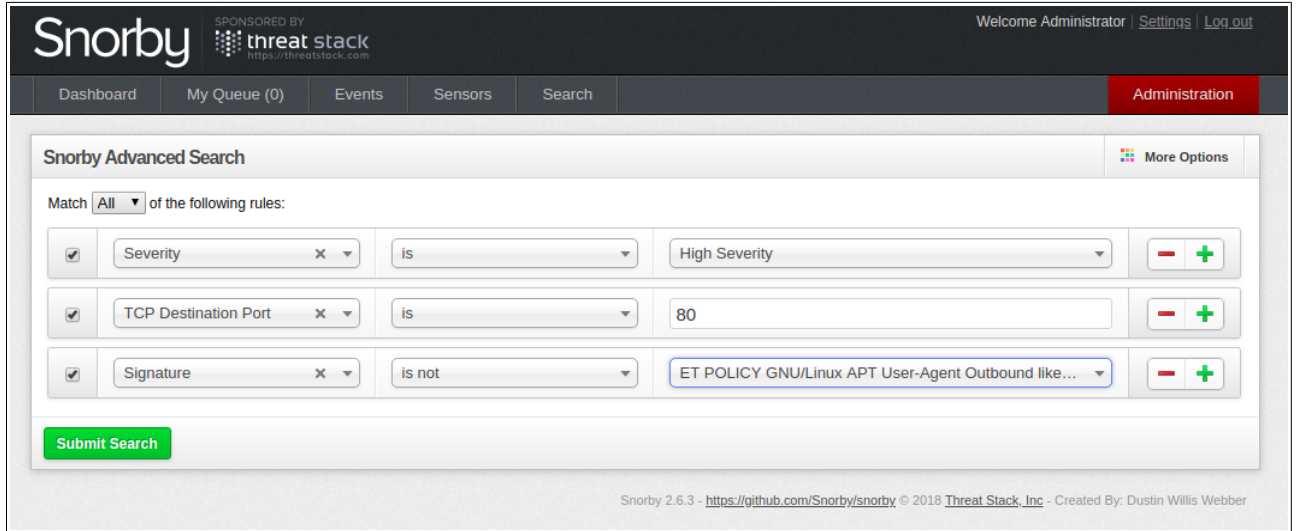
Veri tabanındaki bütün sensorlerin yer aldığı alandır. Bu alandaki View events butonu kullanılarak sadece ilgili sensor'un uyarı kayıtlarına ulaşılabilir.



ID	Name	Hostname	Interface	Last Event	Event Count	Event %
1	Click To Change Me	ahtapot	nfql	06/06/2018 2:36 AM	16,171	100.00%

Search

Bu bölümde veri tabanındaki uyarılar arasından birden fazla arama kriteri verilerek arama yapılabilir.



Match **All** of the following rules:

<input checked="" type="checkbox"/>	Severity	is	High Severity	-	+
<input checked="" type="checkbox"/>	TCP Destination Port	is	80	-	+
<input checked="" type="checkbox"/>	Signature	is not	ET POLICY GNU/Linux APT User-Agent Outbound like...	-	+

Submit Search

Servis Ayar Dosyası: /etc/default/snorby

Uygulama Dizini : /opt/snorby/

Ayar Dosyası: /opt/snorby/config/snorby_config.yml

Veritabanı Ayar Dosyası: /opt/snorby/config/database.yml

İlk veri tabanı oluşturma dosyası: /opt/snorby/db/seeds.rb

Start-Stop Script: /opt/snorby/snorby.init

Bu script'i systemd servis'i kullanıyor. /lib/systemd/system/snorby.service

4- Barnyard2 Kurulum ve Kullanımı

Barnyard2 kurulmadan önce mysql, snorby ve suricata kurulmuş olmalıdır. Ayar dosyasındaki alanlar düzenlenir ve ansible playbook çalıştırılır.

barnyard2_interface: Kayıtlarda görünmesi için gerekli interface adı girilir.

barnyard2_mysql_user: Barnyard2 yazılımının **snorby** database'ine erişim izni olan **mysql** kullanıcı adı girilir.

barnyard2_mysql_password: Mysql kullanıcısının parola bilgisi girilir.

barnyard2_host: Mysql sunucusunun IP adresi yazılır.

barnyard2_sensor_name: Aynı mysql sunucusunda birden fazla noktadan barnyard2 erişimi var ise buraya farklı bir isim yazılır. Snorby üzerinde de bu şekilde ayırt edilecektir.

Ayar Dosyası: /etc/ansible/roles/barnyard2/vars/main.yml

Parametreler:

barnyard2_interface: enp0s3

barnyard2_mysql_user: root

barnyard2_mysql_password: root

barnyard2_host: 127.0.0.1

barnyard2_sensor_name: ids_1

Komut: ansible-playbook /etc/ansible/playbooks/barnyard2.yml

Ayar Dosyası: /etc/barnyard2.conf

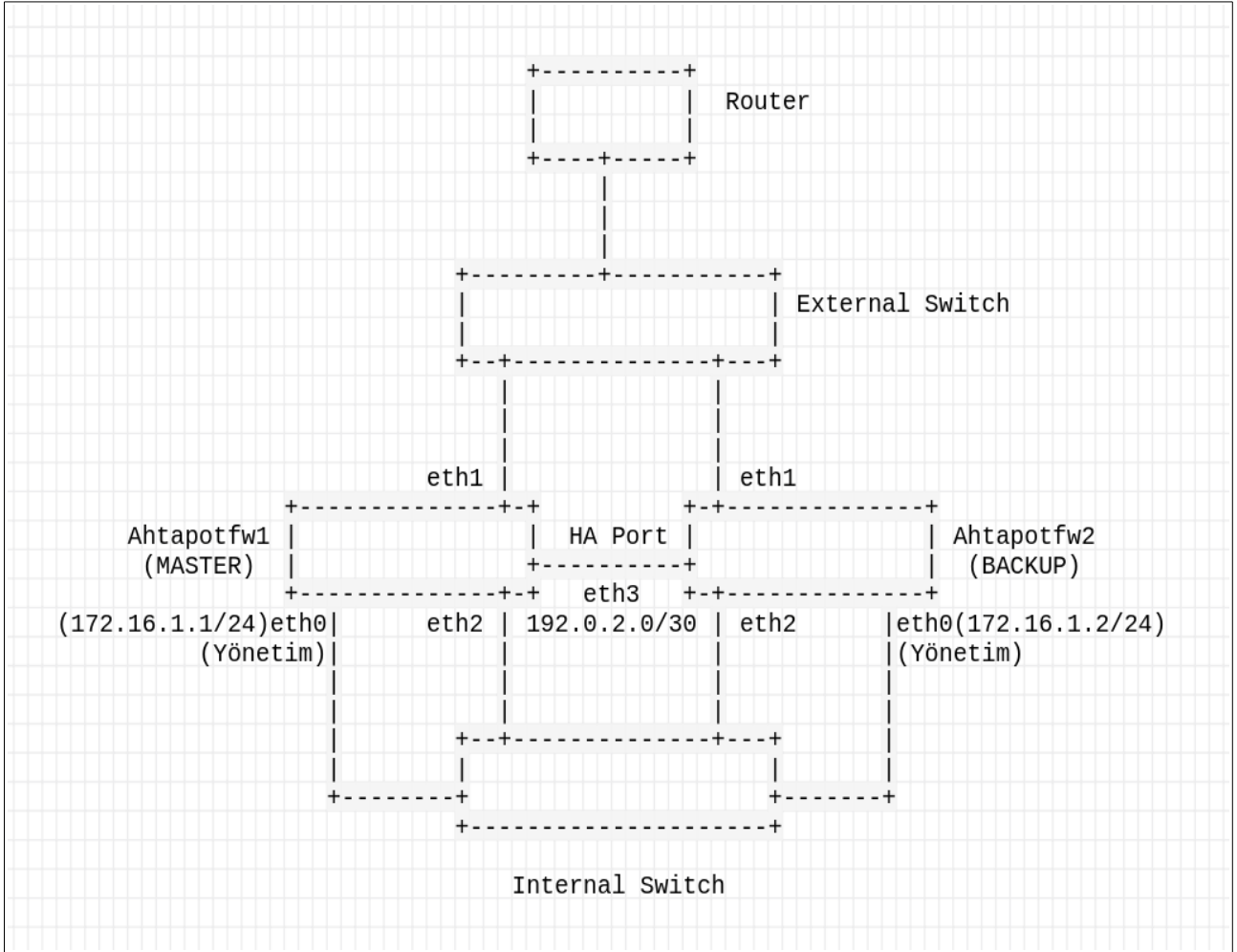
Servis Ayar Dosyası: /etc/default/barnyard2

5- Yedekli Kurulum

Yedekli Saldırı tespit ve önleme sistemi kurulumu NAT/Router ve Bridge mod olarak yapılabilir. NAT/Router mod için sistem var olan [doküman](#) takip edilir.

Bridge Mode Yedekli yapılandırma için aşağıdaki keepalived Ayar dosyası uygulanır ve Bridge interface yapılandırması düzenlenir. Bridge mode yedekli kurulumda iki cihaz arasında Özel bir HA port kullanılmalıdır. Yönetim poru ile sistem toplamda 4 interface ile yapılandırılmalıdır.

Örnek Topoloji



MASTER cihaz keepalived yapılandırması

```
global_defs {
    router_id ahtapot1
}

vrrp_instance ahtapot {
    interface eth3 #HA interface
    state MASTER
    virtual_router_id 52
    priority 200
    authentication {
        auth_type PASS
        auth_pass ahtapot
    }

    #Bridge üyesi fiziksel interface'ler
    track_interface {
        eth1
        eth2
    }
}
```

MASTER cihaz Bridge interface yapılandırması.

[Bridge Mod IPS](#) bölümündeki bridge interface yapılandırması uygulanır. Bu yapılandırmaya ek olarak aşağıdaki parametre eklenir.

Bu ayar sadece master cihazda yapılır. Backup olan cihazda böyle bir yapılandırma **yapılmaz**.

```
#High Availability config
bridge_bridgeprio 30000 # Only Master Node
```

MASTER HA Port Ayarları

```
auto eth3
iface eth3 inet static
    address 192.0.2.1
    netmask 255.255.255.252
```

BACKUP cihaz keepalived yapılandırması

```
global_defs {
    router_id ahtapot2
}

vrrp_instance ahtapot {
    interface eth3 #HA interface
    state BACKUP
    virtual_router_id 52
    priority 100
    authentication {
        auth_type PASS
        auth_pass ahtapot
    }

    #Bridge üyesi fiziksel interface'ler
    track_interface {
        eth1
        eth2
    }
}
```

BACKUP HA Port Ayarları

```
auto eth3
iface eth3 inet static
    address 192.0.2.2
    netmask 255.255.255.252
```

HA portlar her iki cihazda da UP edilir.

```
ifup eth3
```

Diğer kısımlar NAT/Router mode yapılandırma ve test prosedürü ile aynıdır. [Link](#)