

IPS E PENETRATION TEST

INDICE

1. INTRODUZIONE	2
1.1. COS'È VMWARE E VSPHERE	2
1.2. CONFIGURAZIONE INIZIALE	2
1.3. FIREWALL	2
1.4. SERVIZI ATTIVI SULLA MACCHINA	3
1.5. COS'È SNORT	3
1.6. INSTALLAZIONE E CONFIGURAZIONE	4
1.7. TEST PRELIMINARE	4
2. TENTATIVO 2	4
2.1. INSTALLAZIONE E CONFIGURAZIONE	4
2.2. CREAZIONE DI WEB SERVER NELLA DMZ	5
2.3. UN CINESE IN CONNESSIONE CON SAMBA	5
2.4. INSTALLAZIONE KALI E OPENVAS	5
2.5. METASPLOIT E ATTACCO DOS	5
2.6. IPS	5
2.7. FILIPPO LAVORA SU IPS CON FWSNORT	5
2.8. CROWDSEC	6
2.9. TOMCAT BUCATO COME LA MERDA	6
2.10. FILIPPO BLOCCA L'IP DI DAVIDE CON CROWDSEC	8

1. INTRODUZIONE

1.1. COS'È VMWARE E VSPHERE. VMware è un software che permette di creare macchine virtuali, può essere utilizzato per creare un ambiente per effettuare test di sicurezza. L'intera struttura virtuale viene gestita tramite browser grazie allo strumento VSphere Client. Per questa simulazione la versione VMware utilizzata è la 5.5.

1.2. CONFIGURAZIONE INIZIALE. La rete ha questa struttura:

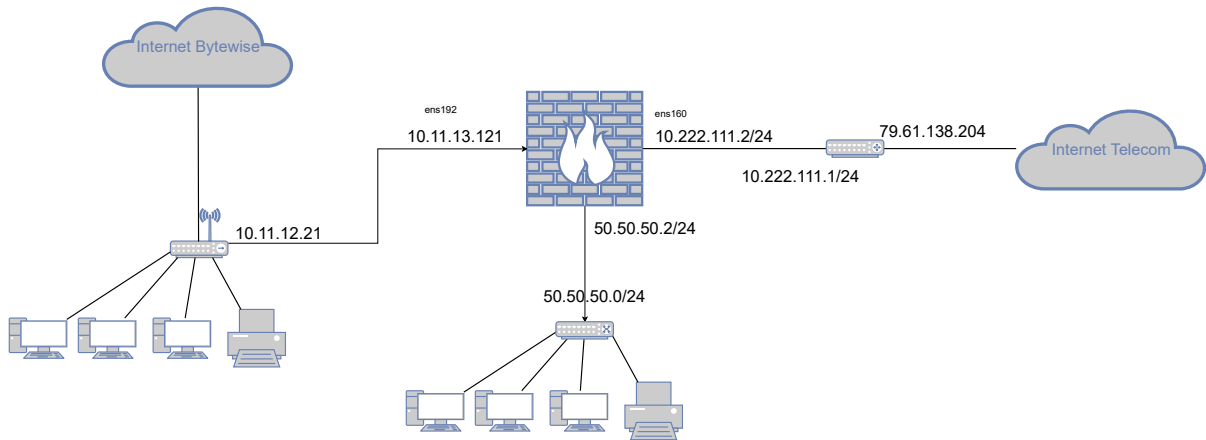


FIGURA 1.1. La rete

Un firewall con tre schede di rete:

- la 10.11.13.121 che si affaccia sulla rete bytewise con default gateway 10.11.12.21. L'interfaccia è ens192.
- la 10.222.111.2 che si affaccia sulla rete telecom con default gateway 10.222.111.1. Questo è il router che viene usato dal firewall quando l'host da raggiungere non è direttamente connesso. L'interfaccia è ens160.
- la 50.50.50.1 sulla lan 50.50.50.0/24 (*sul disegno compare l'IP sbagliato). L'interfaccia è ens224.
- la vm è visibile dall'esterno tramite l'IP pubblico 79.61.138.204

La macchina virtuale che ospita il **firewall** è stata installata con questa configurazione:

Le risorse assegnate alla vm sono:

- 8 core
- 32 GB ram
- 250 GB hdd

1.3. FIREWALL. Il firewall viene configurato manualmente, quindi è stato disabilitato firewalld (systemctl disable firewalld --now) e SELinux.

Il firewall si trova in /root/fwdir.

Algoritmo 1 vi /root/fwdir/flush

```
#!/bin/bash

iptables -F #flush delle catene di forward e input output
iptables -F -t nat #flush delle catene di pre-post routing
iptables -X #delete catene di forward

# politica
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
```

Algoritmo 2 vi /root/fwdir/fw

```
#!/bin/bash

#----- FLUSH FIREWALL -----#
/root/fwdir/flush
echo 1 > /proc/sys/net/ipv4/ip_forward

#----- POLICY -----#
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

#----- FORWARD FILE -----#
/root/myfirewalldir/myfirewallForward

#----- I/O FILE -----#
/root/myfirewalldir/myfirewall_IO
service iptables save
```

Questo file esegue prima il flush, poi i file in cui sono memorizzate le regole di forward e di input output. I file sono visibili su github nel folder personale di Filippo.

1.4. SERVIZI ATTIVI SULLA MACCHINA. I servizi attivi sulla macchina sono:

- Un server http apache con vulnerabilità remote code execution.
- MariaDB Version 5.5.68
 - passwd: mnkjoi09
- Samba Version 4.10.16
 - Guida: <https://linuxize.com/post/how-to-install-and-configure-samba-on-centos-7/>
 - TCP porte 139, 445
 - utente: josh
 - passwd: johnlennon
 - utente: sadmin
 - passwd: ringostarr

1.5. COS'È SNORT. Snort è un software di rilevazione e prevenzione delle intrusioni. I sistemi di rilevazione delle intrusioni (Intrusion Detection Systems, o IDS) individuano pacchetti sospetti, li registrano e ne

segnalano la presenza all'amministratore di rete. La loro naturale evoluzione consiste nei sistemi di prevenzione delle intrusioni (Intrusion Prevention Systems, o IPS), che intraprendono azioni attive sui pacchetti sospetti eliminandoli, resettando connessioni e/o bloccando interamente il traffico proveniente da determinati indirizzi IP.

1.6. INSTALLAZIONE E CONFIGURAZIONE. Snort è stato installato da sorgente:

- snort-2.9.19.tar.gz
- daq-2.0.7.tar.gz (data acquisition library)

La guida è disponibile tramite il link:

<https://upcloud.com/community/tutorials/installing-snort-on-centos/>

1.7. TEST PRELIMINARE. Un test preliminare è stato rilevare tutti i tentativi di connessione sulla porta 22. Bisogna aggiungere una regola nel file `/etc/snort/rules/local.rules`.

La regola è:

- **alert tcp any any -> \$HOME_NET 22 (msg:"ssh test"; sid:10000001; rev:001;)**
 - snort manda un alert per i pacchetti che hanno source-ip: **any**, source-port: **any**, destination-ip: **10.222.111.2** (memorizzato nella variabile `$HOME_NET`), destination-port: **22**

Se un host esterno cerca di connettersi tramite la porta 22, il tentativo viene rilevato da snort.

È possibile far partire un'istanza di snort e far stampare a schermo i log digitando il comando:

- **snort -A console -i ens160 -u snort -g snort -c /etc/snort/snort.conf**
 - Nota: ens160 è l'interfaccia che corrisponde all'ip 10.222.111.2

Grazie ai log di snort e ai log di ssh su `/var/log/secure`, è possibile osservare come ogni secondo ci siano tentativi di login su ssh (ovviamente falliti) dovuti a script automatizzati. Per eliminare il problema, ssh è stato reso disponibile sulla porta 65022.

2. TENTATIVO 2

Il precedente tentativo è andato a mignotte senza nessun motivo apparente, quindi si è reso necessario procedere con una seconda installazione.

2.1. INSTALLAZIONE E CONFIGURAZIONE. Snort è stato installato tramite yum:

- `yum install snort.x86_64`

Per configurare Snort:

Il demone `snortd` legge `/etc/sysconfig/snort`:

- bisogna impostare l'interfaccia (ens160)
- bisogna specificare user (snort) e group (snort)
- binary log attivati

È stato modificato il file `/etc/snort/snort.conf`

- bisogna specificare l'ip address che va protetto
- bisogna specificare il path delle rules
- sfrutta i log dell'API pcap specializzata a catturare il traffico internet
- sfrutta i log unified2 che si occupano di packet logging, alert logging, true unified logging

Per leggere i log:

- `snort -r merged.log`
- comando: `u2spewfoo`

In questo modo è stato possibile attivare il demone `snortd`.

È stato installato PulledPork package che permette di avere accesso alle community rules offerte da snort.

Guida:

<https://support.redborder.com/hc/en-us/articles/209057125-Snort-on-CentOS-7-with-redBorder-Live-Informationi-su-DAQ-pcap>:

<http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node7.html>

2.2. CREAZIONE DI WEB SERVER NELLA DMZ. È stato creato un web server nella DMZ. Questo rende necessario una modifica del firewall.

Sul web server sono installati i servizi:

- Httpd 2.4.6
- Tomcat 7.0.76
- MariaDB
- Samba

2.3. UN CINESE IN CONNESSIONE CON SAMBA. Dopo l'installazione e la configurazione di Samba è stato digitato il comando **netstat -tulnap** ed è stato visto che un IP address sconosciuto (proveniente dalla Cina) è riuscito ad instaurare una connessione con Samba non autorizzata. Questo perchè la versione installata è vecchia e vulnerabile.

2.4. INSTALLAZIONE KALI E OPENVAS. È stata creata una macchina virtuale kali su cui è stato installato openvas. Openvas è un framework che include servizi e strumenti per la scansione e la gestione completa delle vulnerabilità (vulnerability assessment).

Openvas è in grado di fornire un report delle vulnerabilità dell'host, e un primo scan ha segnalato http come il servizio a maggiore rischio (vulnerabilità SSL-TLS):

Una possibile guida per installare Openvas:

<https://www.geeksforgeeks.org/installing-openvas-on-kali-linux/>

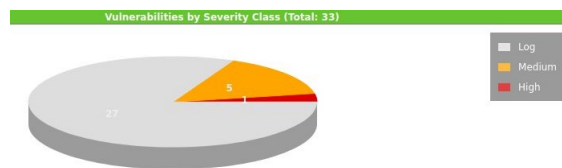


FIGURA 2.1. Vulnerabilities

2.5. METASPLOIT E ATTACCO DOS. Metasploit è un framework open source per lo sviluppo e l'esecuzione di exploits ai danni di un host remoto. Sulla vm Kali è possibile accedere alla console tramite il comando **msfconsole**.

Cercando la vulnerabilità rilevata da openVAS con il comando **search ssl tls** è possibile trovare uno script che effettua un attacco dos ai danni di un host.

Una possibile guida per i comandi base di metasploit:

<https://www.makeuseof.com/beginners-guide-metasploit-kali-linux/>

2.6. IPS. Snort **inline mode** permette di proteggere il sistema. Quando viene generato del traffico:

- legge il contenuto di pacchetti memorizzati in una coda
- legge una serie di regole iptables e in base ad esse decide se lasciarli passare o meno.

La guida usata è:

<http://sublimeroobots.com/2017/06/snort-ips-with-nfq-routing-on-ubuntu/>

Questa guida è stata usata ma non funzionava... in the end è stato sfruttato FWsnort invece di nfqueue

2.7. FILIPPO LAVORA SU IPS CON FWSNORT.

- Snort ha le community rules,
- FWsnort è un progetto opensource che permette di tradurre le regole di snort in iptables eseguendo uno script,

Tra le guide usate:

- (1) <https://www.cipherdyne.org/fwsnort/>
- (2) <https://linux.die.net/man/8/fwsnort>

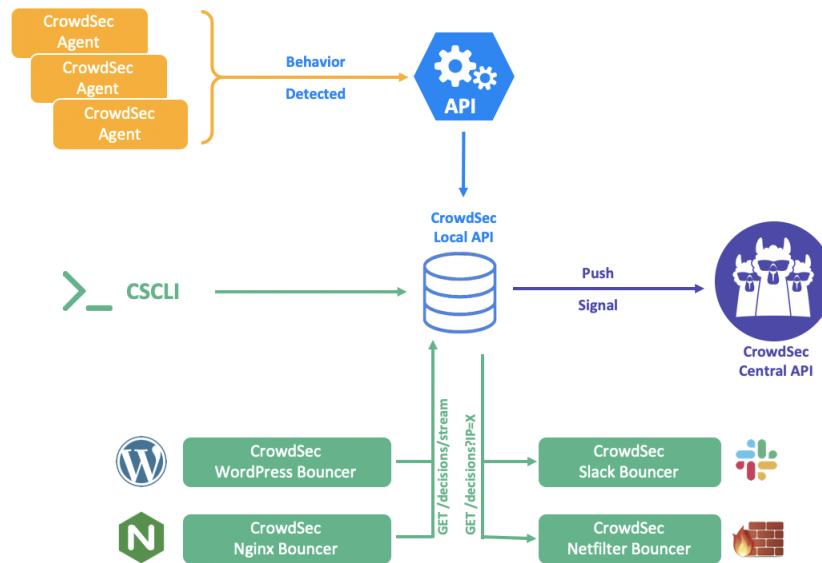


FIGURA 2.2. how does crowdsec work

2.8. **CROWDSEC.** Per configurare crowdsec è stato disabilitato snort e fatto il flush del firewall.

La versione installata è 1.3.4

Guida per installare:

- https://docs.crowdsec.net/docs/getting_started/install_crowdsec/
- <https://opensource.com/article/21/1/crowdsec-rest-api>

2.8.1. *moduli.* È possibile installare dei moduli tramite il comando `cscli install ...`, questi moduli vengono presi in input da un bouncer che genera regole iptables dinamicamente per bloccare il traffico di alcuni ip.

2.8.2. *bouncer.* Per vedere la configurazione del firewall bouncer bisogna eseguire il comando:

```
cat /etc/crowdsec/bouncers/crowdsec-firewall-bouncer.yaml
```

```

root@firewall ~ # cat /etc/crowdsec/bouncers/crowdsec-firewall-bouncer.yaml
mode: iptables
pid_dir: /var/run/
update_frequency: 10s
daemonize: true
log_mode: file
log_dir: /var/log/
log_level: info
log_compression: true
log_max_size: 100
log_max_backups: 3
log_max_age: 30
api_url: http://127.0.0.1:8080/
api_key: e82403a771a4386fbfc730544671643d
insecure_skip_verify: false
disable_ipv6: false
deny_action: DROP
deny_log: false

```

FIGURA 2.3. firewall bounce configuration

2.9. **TOMCAT BUCATO COME LA MERDA.** Probabilmente un mining virus occupava il 22% della cpu circa,

- è stato usato il comando `ps faxw` per cercare il processo da uccidere
- il processo rimandava a una libreria di tomcat

- è stato ucciso il processo
- disabilitato il servizio tomcat

```

root@victim ~ # netstat -tulnap
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN      1333/mysqld
tcp        0      0 0.0.0.0:111            0.0.0.0:*               LISTEN      679/rpcbind
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      963/sshd
tcp        0      0 0.0.0.0:25             0.0.0.0:*               LISTEN      1461/master
tcp        0 113.0 150.50.50.3:42986      146.209.231.154:8081    SYN_SENT    2783/4n4w1h
tcp        0 59.2 150.50.50.3:48098      211.63.39.98:8081      SYN_SENT    2783/4n4w1h
tcp        0 146.0 150.50.50.3:50330      65.179.118.188:8081    SYN_SENT    2783/4n4w1h
tcp        0 9.56 150.50.50.3:55326      97.191.244.124:8081    SYN_SENT    2783/4n4w1h
tcp        0 223.0 150.50.50.3:45630      96.64.11.185:8081      SYN_SENT    2783/4n4w1h
tcp        0 40.1 150.50.50.3:43278      205.213.152.7:8081     SYN_SENT    2783/4n4w1h
tcp        0 212.0 150.50.50.3:45868      93.220.74.78:8081      SYN_SENT    2783/4n4w1h
tcp        0 221.0 150.50.50.3:42314      126.82.213.241:8081    SYN_SENT    2783/4n4w1h
tcp        0 87.2 150.50.50.3:60656      111.39.94.153:8081     SYN_SENT    2783/4n4w1h
tcp        0 199.0 150.50.50.3:56558      104.123.98.251:8081    SYN_SENT    2783/4n4w1h
tcp        0 148.0 150.50.50.3:44362      175.104.59.203:8081    SYN_SENT    2783/4n4w1h
tcp        0 134.0 150.50.50.3:45936      96.219.197.10:8081     SYN_SENT    2783/4n4w1h
tcp        0 147.0 150.50.50.3:55132      131.123.202.62:8081    SYN_SENT    2783/4n4w1h
tcp        0 206.0 150.50.50.3:57236      115.196.236.112:8081   SYN_SENT    2783/4n4w1h
tcp        0 97.6 150.50.50.3:33400      164.135.93.197:8081    SYN_SENT    2783/4n4w1h
tcp        0 38.1 150.50.50.3:57894      83.116.27.121:8081     SYN_SENT    2783/4n4w1h
tcp        0 173.0 150.50.50.3:37122      175.111.205.164:8081   SYN_SENT    2783/4n4w1h
tcp        0 59.2 150.50.50.3:57144      41.255.93.59:8081      SYN_SENT    2783/4n4w1h
tcp        0 89.1 150.50.50.3:46184      188.1.205.187:8081     SYN_SENT    2783/4n4w1h
tcp        0 39.8 150.50.50.3:38372      164.229.15.38:8081     SYN_SENT    2783/4n4w1h
tcp        0 88.1 150.50.50.3:42298      144.202.98.103:8081    SYN_SENT    2783/4n4w1h
tcp        0 86.2 150.50.50.3:39274      84.118.105.9:8081      SYN_SENT    2783/4n4w1h
tcp        0 54.5 150.50.50.3:44298      65.255.161.191:8081    SYN_SENT    2783/4n4w1h

```

FIGURA 2.4. connection attempts

```

May 23 11:41:54 victim server: INFORMAZIONE: Deploying web application archive /var/lib/tomcat/webapps/nwYBEZaI8xq1.war
May 23 11:41:56 victim server: INFORMAZIONE: Deployment of web application archive /var/lib/tomcat/webapps/nwYBEZaI8xq1.war has finished in 1,518 ms
May 23 11:41:56 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/ROOT
May 23 11:41:56 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/ROOT has finished in 496 ms
May 23 11:41:56 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/examples
May 23 11:41:58 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/examples has finished in 1,209 ms
May 23 11:41:58 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/sample
May 23 11:41:58 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/sample has finished in 301 ms
May 23 11:41:58 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/host-manager
May 23 11:41:58 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/host-manager has finished in 301 ms
May 23 11:41:58 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/manager
May 23 11:41:59 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/manager has finished in 527 ms
May 23 11:41:59 victim server: INFORMAZIONE: Deploying web application directory /var/lib/tomcat/webapps/docs
May 23 11:41:59 victim server: INFORMAZIONE: Deployment of web application directory /var/lib/tomcat/webapps/docs has finished in 357 ms
May 23 11:42:01 victim custom: Created alias User Alias of tomcat

```

FIGURA 2.5. cat /var/log/messages | grep tomcat

```

root@victim ~ # cat /var/log/messages | grep deploy
May 22 22:38:16 victim server: mag 22, 2022 10:38:16 PM org.apache.catalina.startup.HostConfig deployWAR
May 22 22:38:17 victim server: mag 22, 2022 10:38:17 PM org.apache.catalina.startup.HostConfig deployWAR
May 22 22:38:51 victim server: mag 22, 2022 10:38:51 PM org.apache.catalina.startup.HostConfig undeploy
May 22 22:38:51 victim server: INFORMAZIONI: Undeploying context [/3bfoy0tv]
May 23 11:41:54 victim server: mag 23, 2022 11:41:54 AM org.apache.catalina.startup.HostConfig deployWAR
May 23 11:41:56 victim server: mag 23, 2022 11:41:56 AM org.apache.catalina.startup.HostConfig deployWAR
May 23 11:41:56 victim server: mag 23, 2022 11:41:56 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:56 victim server: mag 23, 2022 11:41:56 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:56 victim server: mag 23, 2022 11:41:56 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:58 victim server: mag 23, 2022 11:41:58 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:58 victim server: mag 23, 2022 11:41:58 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:58 victim server: mag 23, 2022 11:41:58 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:58 victim server: mag 23, 2022 11:41:58 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:58 victim server: mag 23, 2022 11:41:58 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:59 victim server: mag 23, 2022 11:41:59 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:59 victim server: mag 23, 2022 11:41:59 AM org.apache.catalina.startup.HostConfig deployDirectory
May 23 11:41:59 victim server: mag 23, 2022 11:41:59 AM org.apache.catalina.startup.HostConfig deployDirectory

```

FIGURA 2.6. cat /var/log/messages | grep deploy

2.10. **FILIPPO BLOCCA L'IP DI DAVIDE CON CROWDSEC.** Davide è entrato in ssh con ip 62.19.296.203. È stato possibile rilevare la sessione di root tramite il comando w e bloccare l'ip aggiungendolo alla decisions list di crowdsec

```

root@firewall ~ # w
14:53:06 up 4 days, 22:58, 7 users, load average: 0.01, 0.02, 0.05
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU WHAT
root      pts/1    185.58.120.208  14:20    16:58  0.35s  0.35s -bash
root      pts/0    185.58.120.208  14:18    2.00s  1.33s  0.04s w
root      pts/2    185.58.120.208  14:27    23:54  0.36s  0.36s -bash
root      pts/3    185.58.120.208  10:48    4:00m  0.27s  0.27s -bash
root      pts/4    185.58.120.208  14:36    16:40  0.20s  0.20s -bash
root      pts/5    62.19.196.203   14:52    8.00s  0.21s  0.21s -bash
root      pts/7    185.58.120.208  11:21    16:42  0.73s  0.73s -bash
root@firewall ~ # cscli decisions add --ip 62.19.196.203
INFO[23-05-2022 02:59:54 PM] Decision successfully added

```

FIGURA 2.7. root sessions

```

root@firewall ~ # cscli decisions list
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | SOURCE | SCOPE:VALUE | REASON | ACTION | COUNTRY | AS | EVENTS | EXPIRATION | ALERT ID |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 5351 | cscli | Ip:62.19.196.203 | manual 'ban' from | ban | | | 1 | 3h59m43.278063955s | 3 |
| | | | '022de521adca4405817a0cb9735112fa1Dou3xo1BGXb8R0u' | | | | | | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

FIGURA 2.8. decision list