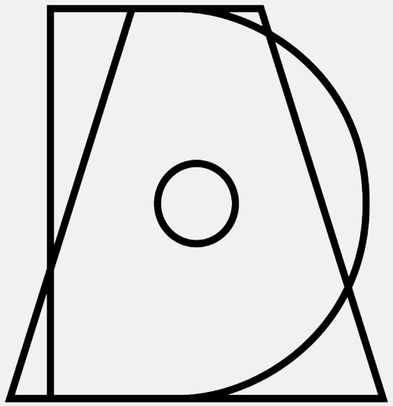
WHAT I LEARNED...
...FROM RUNNING A HONEYPOT



DROID ANDY

security researcher

$WHOAMI

» andi

» owasp newcastle chapter leader

» security researcher

» mobile security


R416
BLC


BMW Motorsport GmbH

WHAT IS A HONEYPOT?

» intentionally vulnerable system

» value lies in being attacked / compromised

» information gathering

» interaction

WHY? (MOTIVATION)

» capture live attacks (playground)

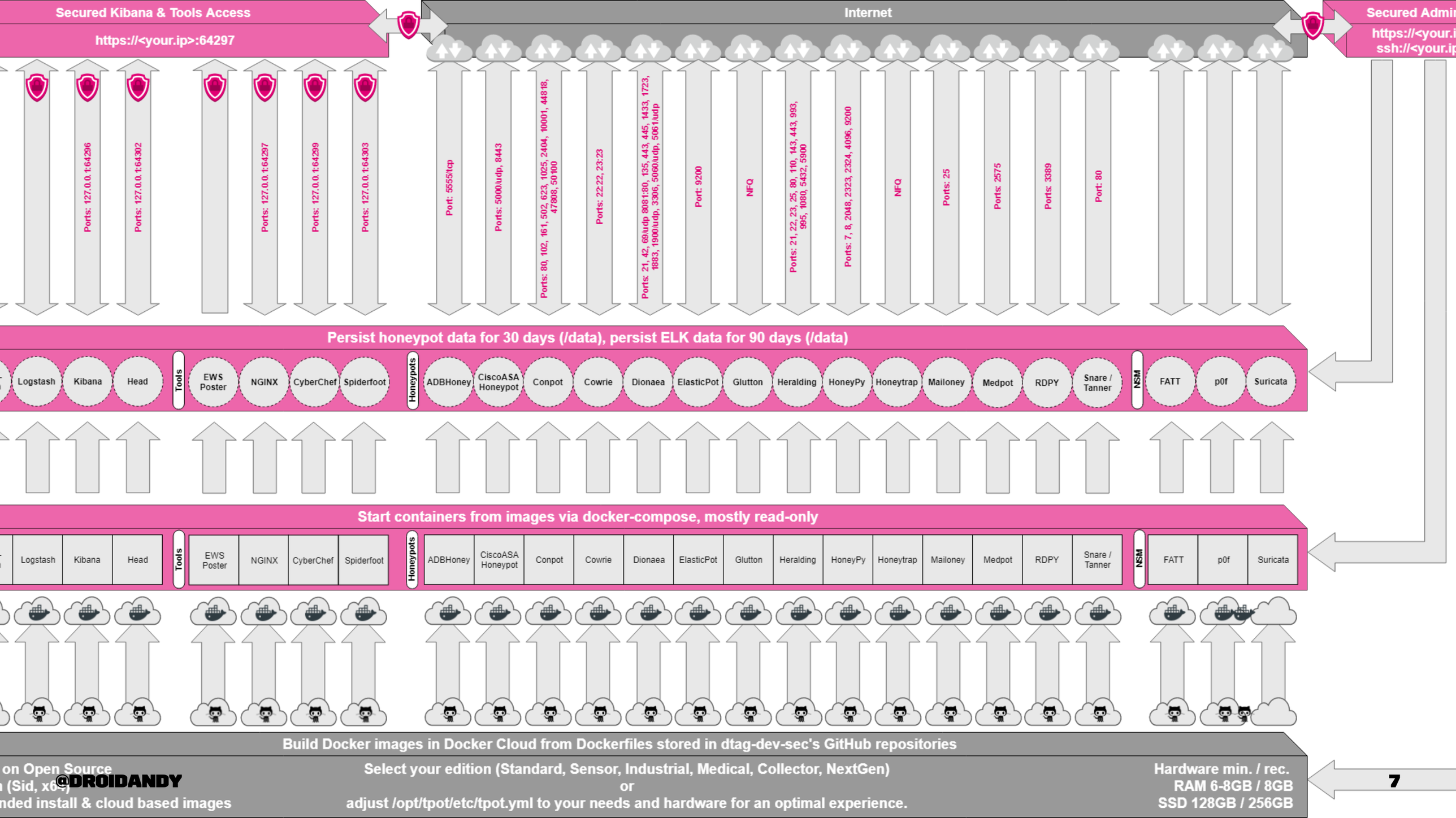
» story

SETUP

» cloud ☁

» docker 🔒

» guide 📖


Secured Kibana & Tools Access
https://<your.ip>:64297
Internet
Secured Admi
https://<your.i
ssh://<your.ip
Ports: 127.0.0.1:64296
Ports: 127.0.0.1:64302
Ports: 127.0.0.1:64297
Ports: 127.0.0.1:64299
Ports: 127.0.0.1:64303
Port: 5555/tcp
Ports: 5000/udp, 8443
Ports: 80, 102, 161, 502, 623, 1025, 2404, 10001, 44818, 47808, 50100
Ports: 22:22, 23:23
Ports: 21, 42, 69/udp 8081:80, 135, 443, 445, 1433, 1723, 1883, 1900/udp, 3306, 5060/udp, 5061/udp
Port: 9200
NFQ
Ports: 21, 22, 23, 25, 80, 110, 143, 443, 993, 995, 1080, 5432, 5900
Ports: 7, 8, 2048, 2323, 2324, 4096, 9200
NFQ
Ports: 25
Ports: 2575
Ports: 3389
Port: 80
Persist honeypot data for 30 days (/data), persist ELK data for 90 days (/data)
Logstash
Kibana
Head
Tools
EWS Poster
NGINX
CyberChef
Spiderfoot
Honeypots
ADBHoney
CiscoASA Honeypot
Conpot
Cowrie
Dionaea
ElasticPot
Glutton
Heralding
HoneyPy
Honeytrap
Mailoney
Medpot
RDPY
Snare / Tanner
NSM
FATT
p0f
Suricata
Start containers from images via docker-compose, mostly read-only
Logstash
Kibana
Head
Tools
EWS Poster
NGINX
CyberChef
Spiderfoot
Honeypots
ADBHoney
CiscoASA Honeypot
Conpot
Cowrie
Dionaea
ElasticPot
Glutton
Heralding
HoneyPy
Honeytrap
Mailoney
Medpot
RDPY
Snare / Tanner
NSM
FATT
p0f
Suricata
Build Docker images in Docker Cloud from Dockerfiles stored in dtag-dev-sec's GitHub repositories
on Open Source
(Sid, x64)
nded install & cloud based images
@DROIDANDY
Select your edition (Standard, Sensor, Industrial, Medical, Collector, NextGen)
or
adjust /opt/tpot/etc/tpot.yml to your needs and hardware for an optimal experience.
Hardware min. / rec.
RAM 6-8GB / 8GB
SSD 128GB / 256GB
7

RESULTS

Focused on 3 honeypots

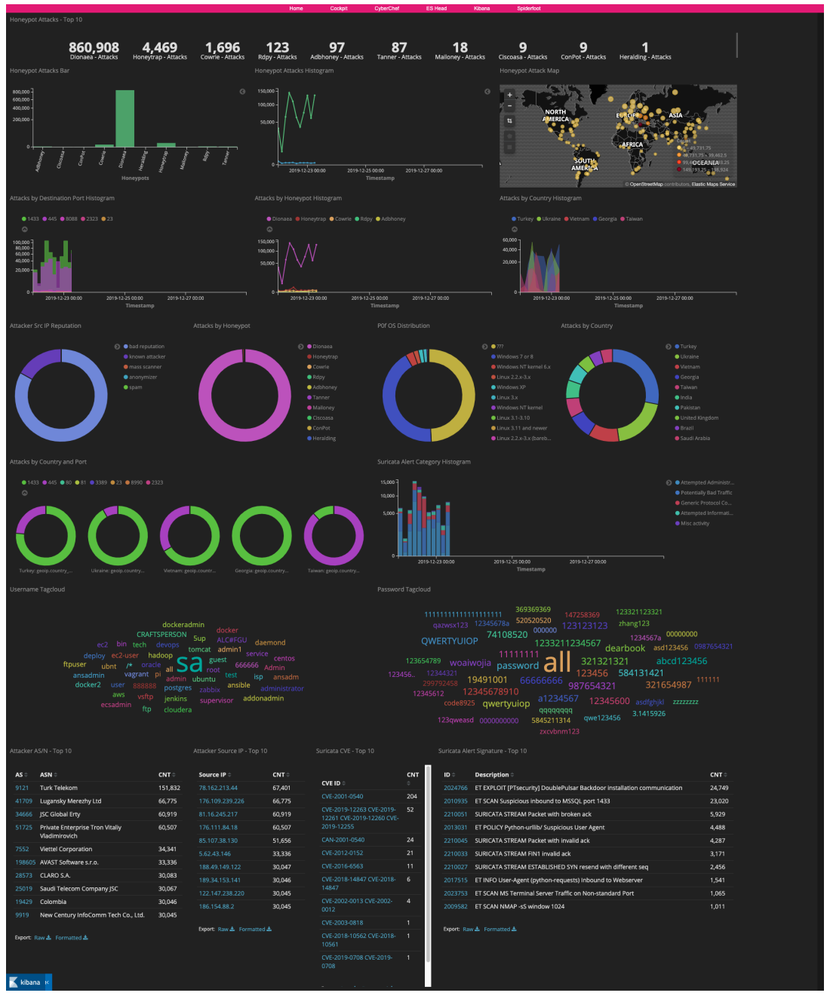
» cowrie (ssh)

» dionaea (ftp)

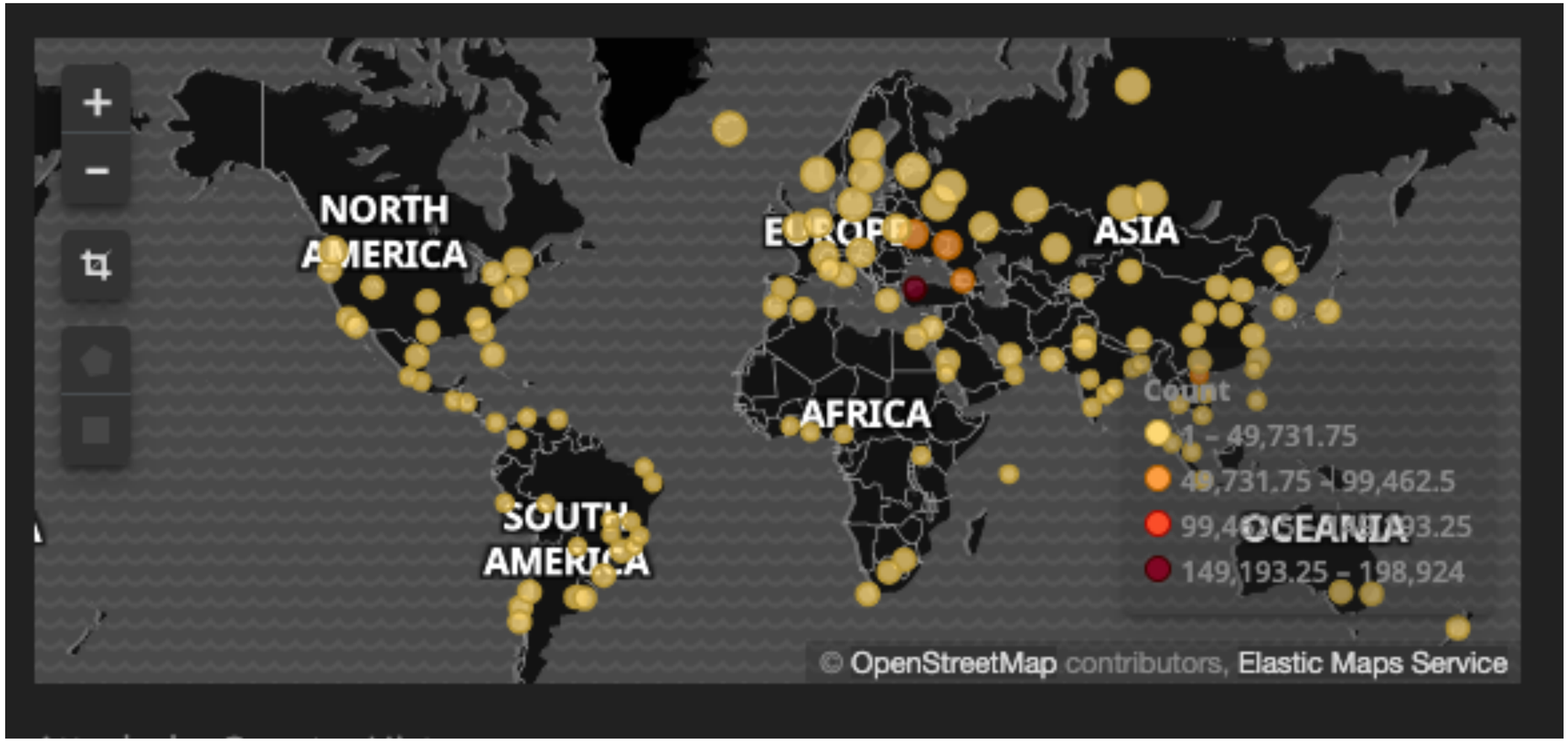
» rdpy (rdp)

» honeytap ???

DASHBOARD


Home
Cockpit
CyberChef
ES Head
Kibana
Spiderfoot
Honeypot Attacks - Top 10
860,908 Dionaea - Attacks
4,469 Honeytrap - Attacks
1,696 Cowrie - Attacks
123 Rdpy - Attacks
97 Adbhoney - Attacks
87 Tanner - Attacks
18 Malloney - Attacks
9 Ciscoasa - Attacks
9 ConPot - Attacks
1 Heralding - Attacks
Honeypot Attacks Bar
Honeypot Attacks Histogram
Honeypot Attack Map
Attacks by Destination Port Histogram
Attacks by Honeypot Histogram
Attacks by Country Histogram
Attacker Src IP Reputation
Attacks by Honeypot
PDF OS Distribution
Attacks by Country
Attacks by Country and Port
Suricata Alert Category Histogram
Username Tagcloud
Password Tagcloud
Attacker AS/N - Top 10
Attacker Source IP - Top 10
Suricata CVE - Top 10
Suricata Alert Signature - Top 10
kibana

MAP


NORTH
AMERICA
EUROPE
ASIA
AFRICA
SOUTH
AMERICA
OCEANIA
Count
1 – 49,731.75
49,731.75 – 99,462.5
149,193.25 – 198,924
© OpenStreetMap contributors, Elastic Maps Service

ATTACKS

» Dionaea [860,908]

» Honeytap [4,469]

» Cowrie [1,696]

USERNAMES

» sa [548,418]

» [88]

» root [80]

» admin [24]

» ubuntu [15]

PASSWORDS

» [334]

» 66666666 [88]

» 123456 [87]

» password [87]

» 321321321 [86]

» 987654321 [86]

SCRIPTS

```
» unset HISTFILE; unset SAVEHIST

» echo "unset HISTFILE; unset SAVEHIST" >> ~/.bashrc

» wget exploit.x86

» curl mussolini.pl

» history -c
```

MISTAKES

FIREWALL RULES

Name	Type	Description	Targets	Filters	Protocols/ports	Action	Priority
default-allow-http	Ingress	http-server	Apply to all	IP ranges: 0.0.0.0/0	tcp:80	Allow	1000
default-allow-https	Ingress	https-server	Apply to all	IP ranges: 0.0.0.0/0	tcp:443	Allow	1000
default-allow-icmp	Ingress	Allow ICMP from anywhere	Apply to all	IP ranges: 0.0.0.0/0	icmp	Allow	1000
default-allow-ssh	Ingress	Allow SSH from anywhere	Apply to all	IP ranges: 0.0.0.0/0	tcp:22	Allow	1000

~~MISTAKES~~ LEARNING POINTS / CHALLENGES

» firewall rules

» gcp

» following the guide

CONCLUSION

» mostly automated attacks

» compromise and then wait

» interesting to learn what is happening

FUTURE WORK

» Customised

» Deployment

» Automate

OH AND...


Andrew Waite
@Infosanity
Trying to kick 2020 off on the right foot, minimal viable ec2 UserData script to automate Cowrie honeypot deployment.
blog.infosanity.co.uk/?p=1397
9:01 PM · Jan 1, 2020 · Twitter Web App

LINKS

https://medium.com/@Stephen_Chap/deploying-monitoring-honeypots-on-gcp-with-kibana-899fef6fdf76

https://github.com/dtag-dev-sec/tpotce