THE POWER OF INTELLIGENT FLOWS

REAL-TIME IOT BOTNET CLASSIFICATION WITH APACHE NIFI

Andre Fucs de Miranda - Fluenda
Andy LoPresto - Hortonworks

Agenda

- Who are the two blokes in front of you
- A brief prologue
- Logs! Logs! Logs!
- The challenge
- The solution
- Wrapping up

Who are the two blokes in front of you

Andre Fucs de Miranda

- Nearly 20 years working with ~~information~~ cyber security

- Logging aficionado (i.e. security data engineer)

- Apache NiFi PMC Member

@trixpan @trixpan

Andy LoPresto

- Financial security & device firmware at Apple, TigerText, etc.

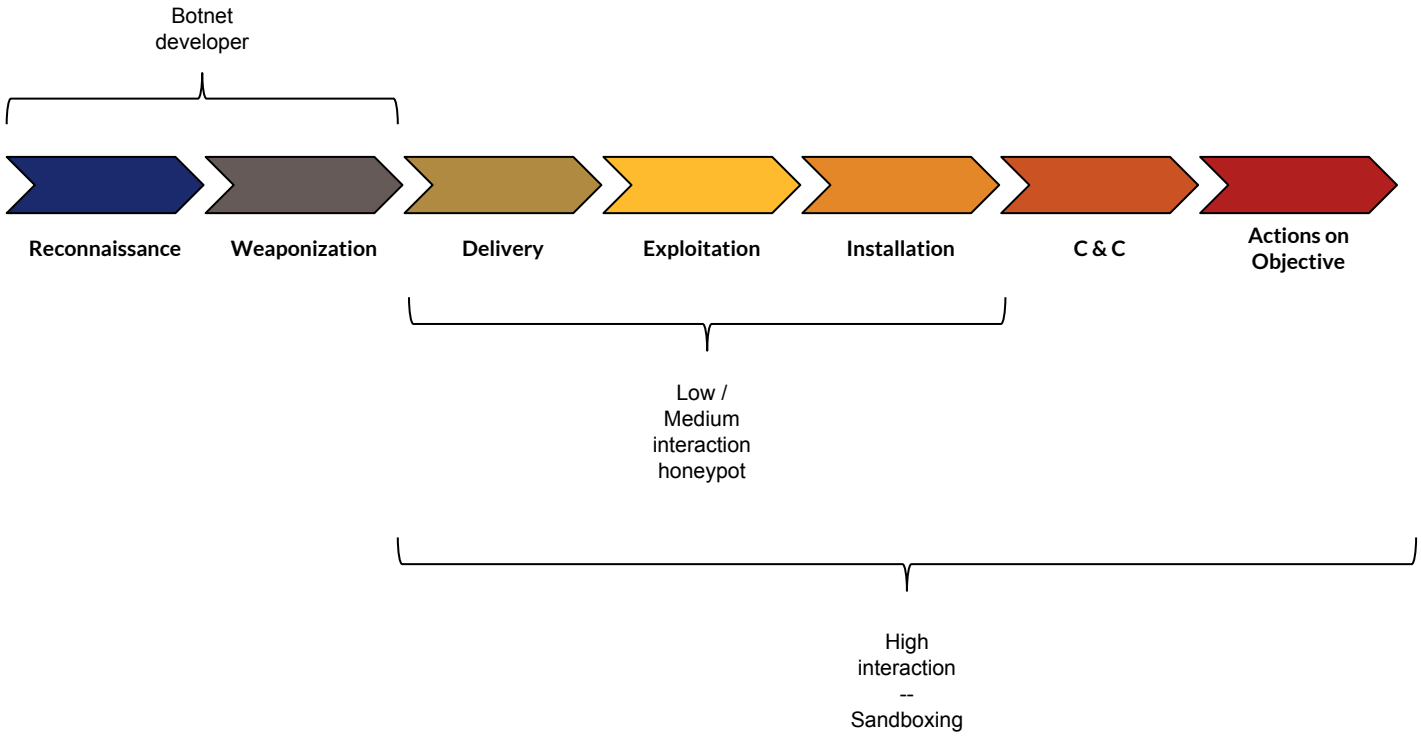
- PII, PCI & EPHI encryption & cracking

- Apache NiFi PMC Member

@yolopey @alopresto

A brief prologue

The Botnet Kill Chain & the Honeypot



The Botnet Kill Chain & the Honeypot

Step 1
Logon to system

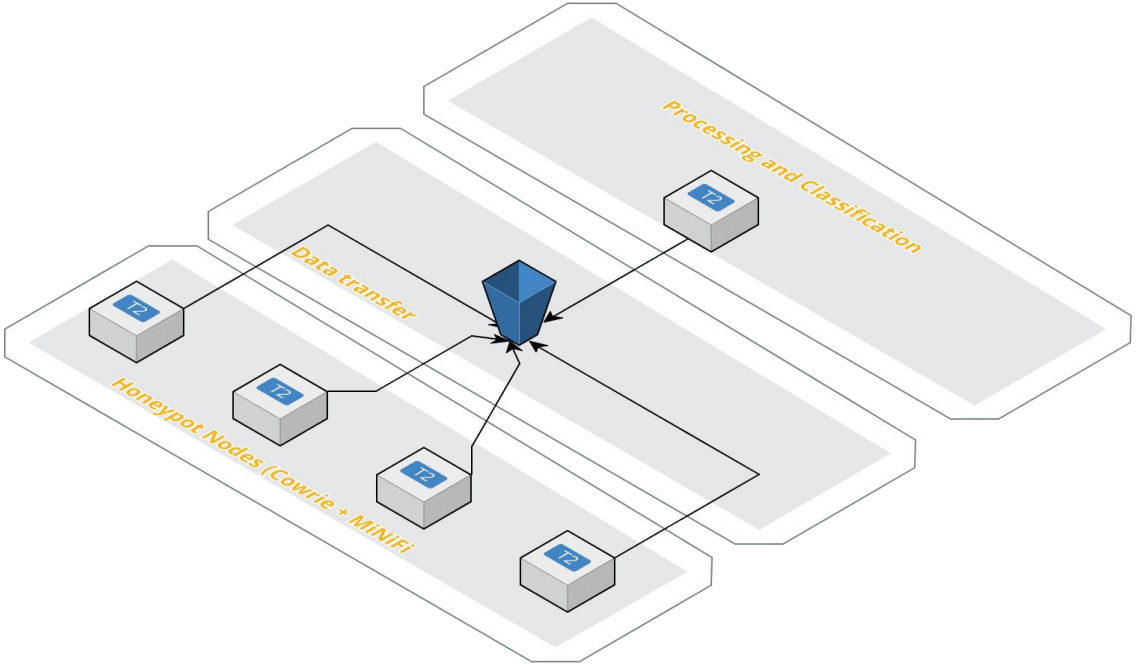
Step 2
Execute predefined sequence of commands

Step 3
Try to install some sort of persistence

The demo environment

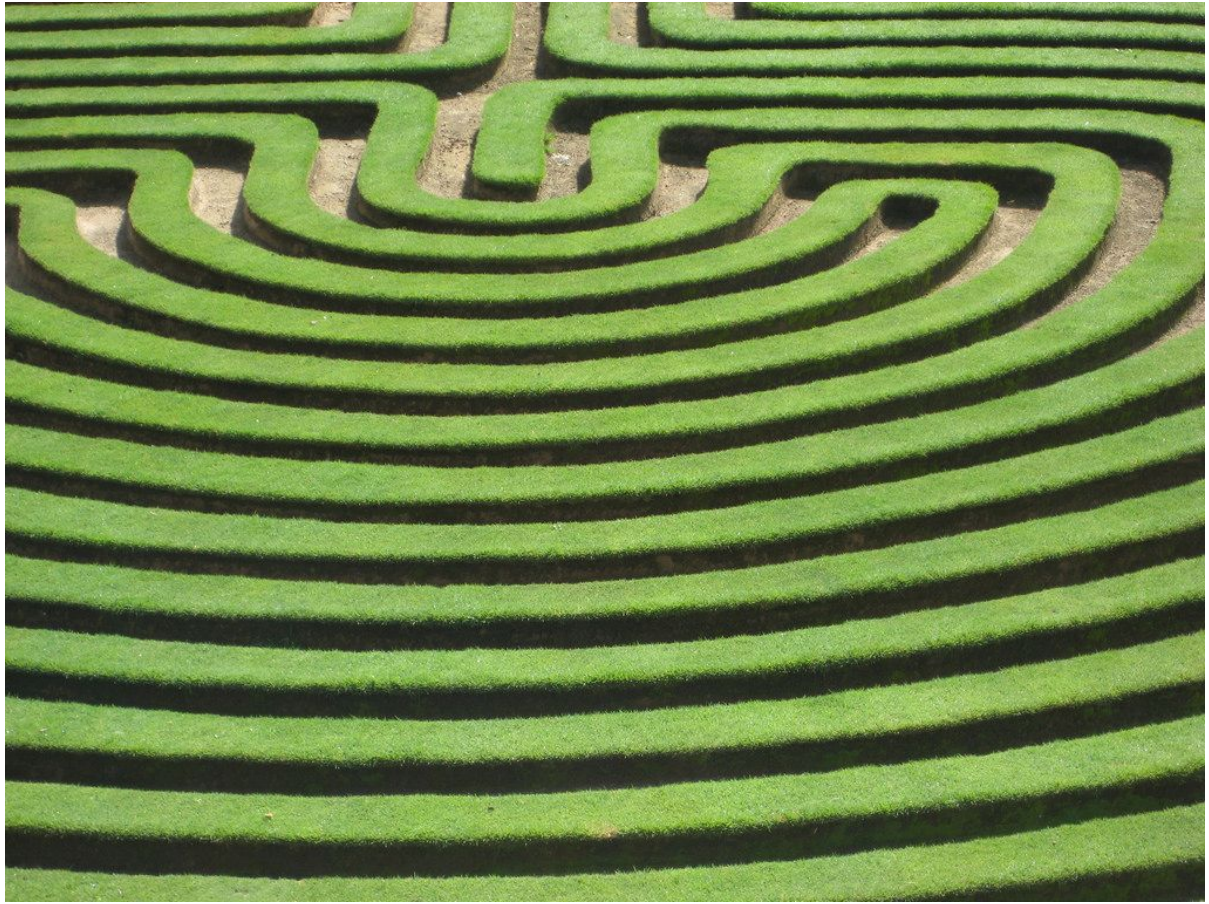
- A handful of EC2 instances running:
  - Cowrie - Medium interaction SSH / Telnet honeypot
  - MiNiFi

- An EC2 instance running:
  - NiFi 1.3.0 (with security enabled)



Flow Design Approach

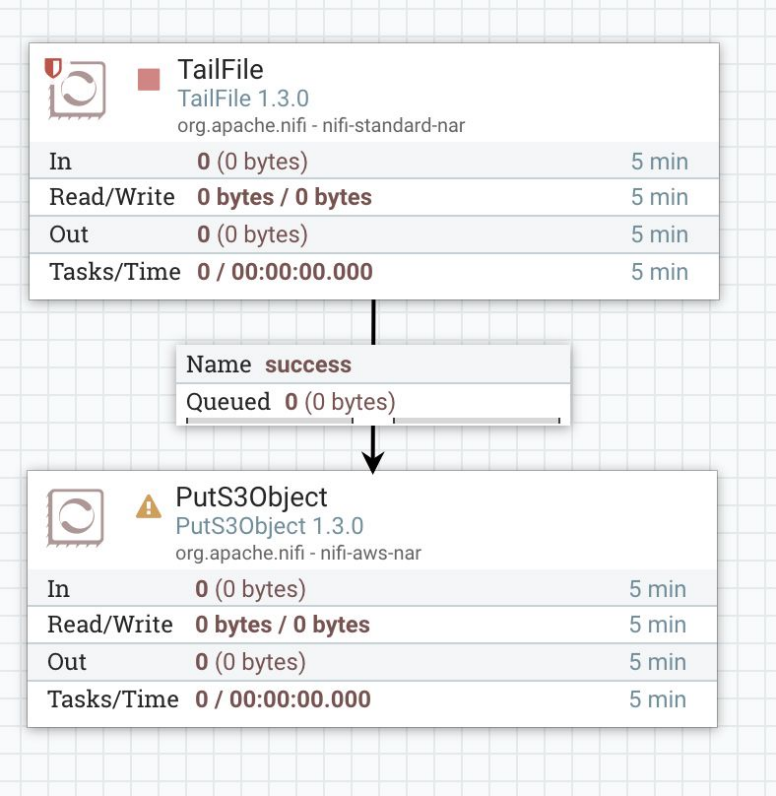
- Don’t be prescriptive
- Treat everything as data
- Don’t be limited by prior expectations
- *Start from the end*



Logs! Logs! Logs!

MiNiFi Process Group

- Tailing a log file being written by cowrie
- Pushing to Amazon S3
  - Could stream via NiFi Site to Site
  - MiNiFi extensibility
  - Shows multiple capabilities
  - Decoupled/no lock in



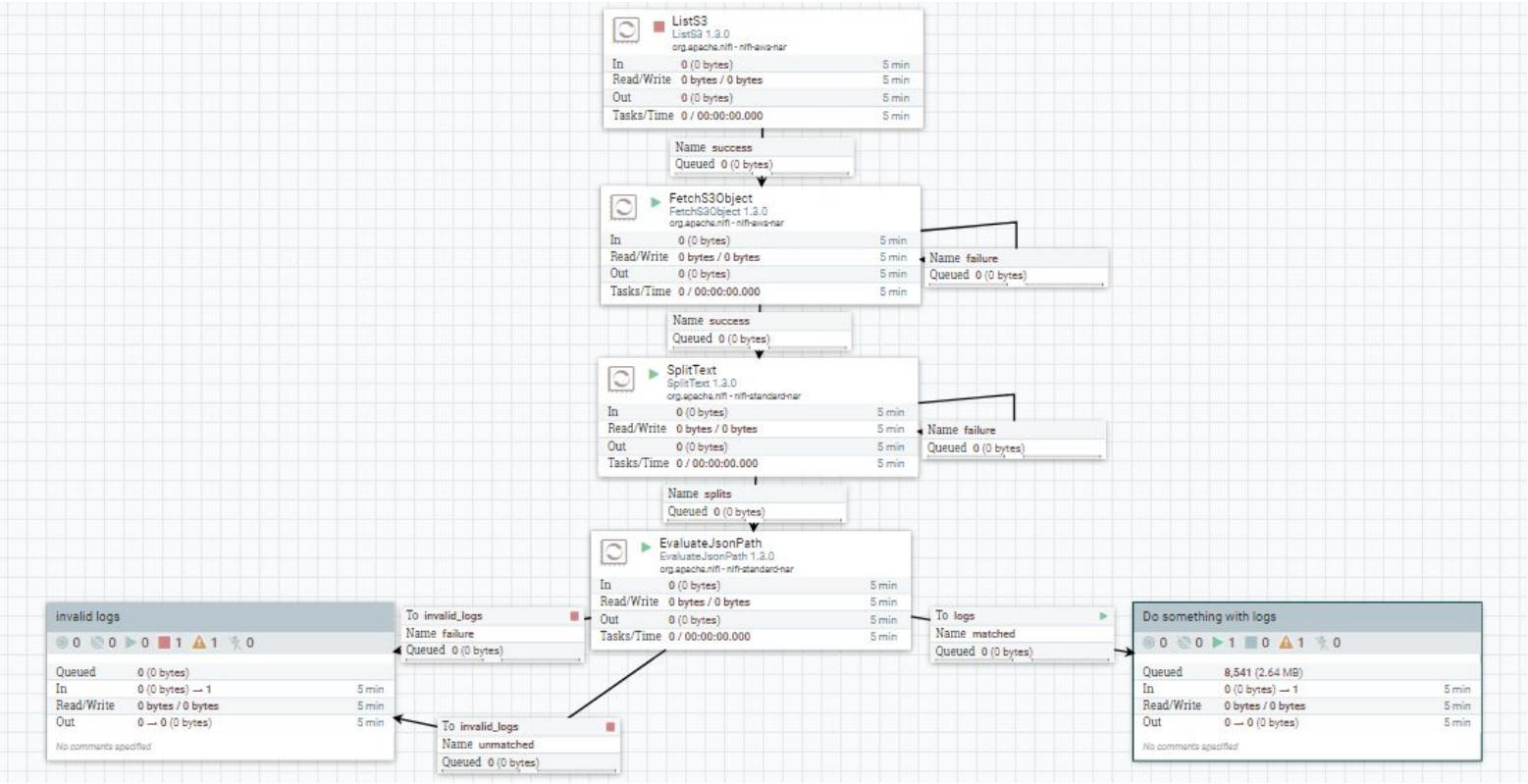
The data being ingested

- Cowrie logs include:
  - Username / Password
  - Commands executed (and parameters)
  - Files downloaded
- Single line JSON entries
  - Easy to parse
  - Textbook machine readable log format
    - Perfect match to NiFi processors such as:
      - SplitText
      - EvaluateJSONPath

Cowrie log example

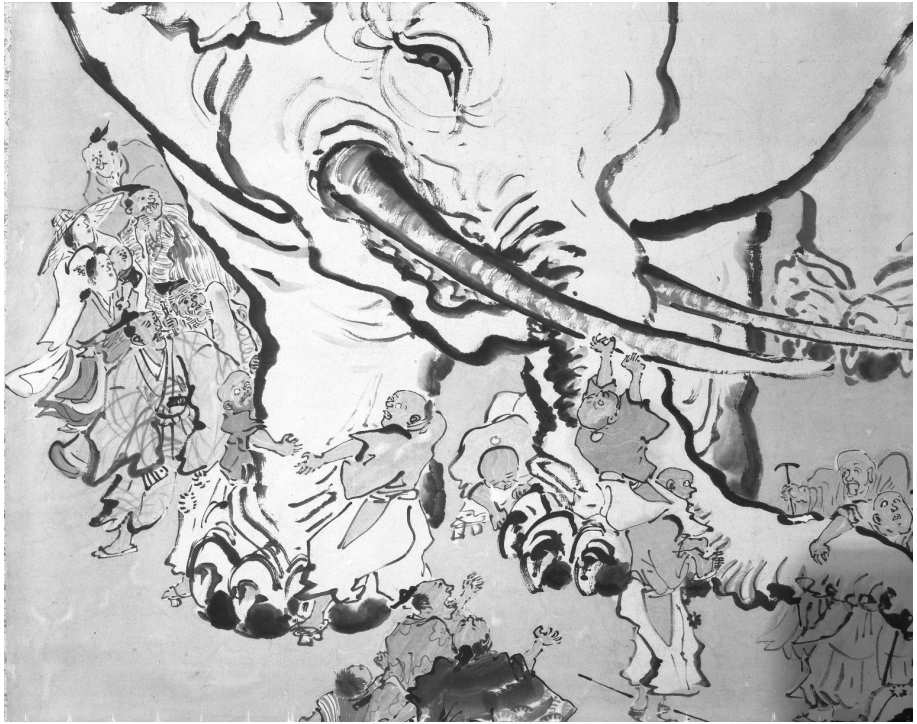

```
{"eventid": "cowrie.log.closed", "timestamp": "2017-09-20T00:37:24.713248Z", "message": "Closing TTY Log: log/tty/20170920-003522-None-2128i.log after 122 se
{"eventid": "cowrie.session.closed", "timestamp": "2017-09-20T00:37:24.719938Z", "message": "Connection lost after 127 seconds", "system": "CowrieTelnetTrans
{"eventid": "cowrie.session.connect", "timestamp": "2017-09-20T00:35:17.055727Z", "session": "82f705bca22c", "message": "New connection: 51.15.221.208:42242
{"eventid": "cowrie.login.success", "username": "root", "timestamp": "2017-09-20T00:35:21.883371Z", "message": "login attempt [root/1234] succeeded", "system
{"eventid": "cowrie.log.open", "timestamp": "2017-09-20T00:35:22.209562Z", "message": "Opening TTY Log: log/tty/20170920-003522-None-2128i.log", "ttylog": "l
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.290251Z", "message": "CMD: enable", "system": "CowrieTelnetTransport,2128,51.15.221.208
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.291474Z", "message": "Command found: enable ", "system": "CowrieTelnetTransport,2128,
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.368973Z", "message": "CMD: shell", "system": "CowrieTelnetTransport,2128,51.15.221.208"
{"eventid": "cowrie.command.failed", "timestamp": "2017-09-20T00:35:22.370123Z", "message": "Command not found: shell", "system": "CowrieTelnetTransport,2128
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.370877Z", "message": "CMD: sh", "system": "CowrieTelnetTransport,2128,51.15.221.208", "
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.371830Z", "message": "Command found: sh ", "system": "CowrieTelnetTransport,2128,51.1
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.485810Z", "message": "CMD: /bin/busybox ECCHI", "system": "CowrieTelnetTransport,2128,5
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.486541Z", "message": "Command found: /bin/busybox ECCHI", "system": "CowrieTelnetTran
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.568321Z", "message": "CMD: /bin/busybox ps; /bin/busybox ECCHI", "system": "CowrieTelne
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.569059Z", "message": "Command found: /bin/busybox ps", "system": "CowrieTelnetTranspo
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.570106Z", "message": "Command found: ps", "system": "CowrieTelnetTransport,2128,51.15
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.571035Z", "message": "Command found: /bin/busybox ECCHI", "system": "CowrieTelnetTran
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.655910Z", "message": "CMD: /bin/busybox cat /proc/mounts; /bin/busybox ECCHI", "system"
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.656838Z", "message": "Command found: /bin/busybox cat /proc/mounts", "system": "Cowri
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.657870Z", "message": "Command found: cat /proc/mounts", "system": "CowrieTelnetTransp
{"eventid": "cowrie.command.success", "timestamp": "2017-09-20T00:35:22.658683Z", "message": "Command found: /bin/busybox ECCHI", "system": "CowrieTelnetTran
{"eventid": "cowrie.command.input", "timestamp": "2017-09-20T00:35:22.746314Z", "message": "CMD: /bin/busybox echo -e '\\x6b\\x61\\x6d\\x69' > /.nippon; /bin
```

Simple Cowrie log ingestion with NiFi



The challenge

The challenge



The challenge

- Logs in isolation rarely will provide the reader with a meaningful view over what is happening

- Verbosity means sensors generate lots of “events”, but who cares about a bot trying to `cat /proc/mounts` ?

- Bots use semi-random values to make detection more difficult.

The solution

Logs are data too...

View as: formatted


```
113     "sensor" : "a927e8b28666"
114   }, {
115     "eventid" : "cowrie.command.input",
116     "timestamp" : "2017-09-18T11:45:25.025835Z",
117     "message" : "CMD: cat /proc/mounts; /bin/busybox LSUCT",
118     "system" : "CowrieTelnetTransport,787,121.237.129.163",
119     "isError" : 0,
120     "src_ip" : "121.237.129.163",
121     "session" : "21caf72c6358",
122     "input" : "cat /proc/mounts; /bin/busybox LSUCT",
123     "sensor" : "a927e8b28666"
124   }, {
125     "eventid" : "cowrie.command.success",
126     "timestamp" : "2017-09-18T11:45:25.027153Z",
127     "message" : "Command found: cat /proc/mounts",
128     "system" : "CowrieTelnetTransport,787,121.237.129.163",
129     "isError" : 0,
130     "src_ip" : "121.237.129.163",
131     "session" : "21caf72c6358",
132     "input" : "cat /proc/mounts",
133     "sensor" : "a927e8b28666"
134   }, {
135     "eventid" : "cowrie.command.success",
136     "timestamp" : "2017-09-18T11:45:25.028091Z",
137     "message" : "Command found: /bin/busybox LSUCT",
138     "system" : "CowrieTelnetTransport,787,121.237.129.163",
139     "isError" : 0,
140     "src_ip" : "121.237.129.163",
141     "session" : "21caf72c6358",
142     "input" : "/bin/busybox LSUCT",
143     "sensor" : "a927e8b28666"
144   }, {
145     "eventid" : "cowrie.command.input",
146     "timestamp" : "2017-09-18T11:45:25.367150Z",
147     "message" : "CMD: cd /dev/shm; cat .s || cp /bin/echo .s; /bin/busybox LSUCT",
148     "system" : "CowrieTelnetTransport,787,121.237.129.163",
149     "isError" : 0,
150     "src_ip" : "121.237.129.163",
151     "session" : "21caf72c6358",
152     "input" : "cd /dev/shm; cat .s || cp /bin/echo .s; /bin/busybox LSUCT",
153     "sensor" : "a927e8b28666"
154   }, {
155     "eventid" : "cowrie.command.success",
```


View as: formatted


```
147   }, {
148     "eventid" : "cowrie.command.input",
149     "timestamp" : "2017-09-17T04:06:39.670673Z",
150     "message" : "CMD: cat /proc/mounts; /bin/busybox XUSRH",
151     "system" : "CowrieTelnetTransport,93,94.51.110.74",
152     "isError" : 0,
153     "src_ip" : "94.51.110.74",
154     "session" : "4c047bbc016c",
155     "input" : "cat /proc/mounts; /bin/busybox XUSRH",
156     "sensor" : "a927e8b28666"
157   }, {
158     "eventid" : "cowrie.command.success",
159     "timestamp" : "2017-09-17T04:06:39.672190Z",
160     "message" : "Command found: cat /proc/mounts",
161     "system" : "CowrieTelnetTransport,93,94.51.110.74",
162     "isError" : 0,
163     "src_ip" : "94.51.110.74",
164     "session" : "4c047bbc016c",
165     "input" : "cat /proc/mounts",
166     "sensor" : "a927e8b28666"
167   }, {
168     "eventid" : "cowrie.command.success",
169     "timestamp" : "2017-09-17T04:06:39.673206Z",
170     "message" : "Command found: /bin/busybox XUSRH",
171     "system" : "CowrieTelnetTransport,93,94.51.110.74",
172     "isError" : 0,
173     "src_ip" : "94.51.110.74",
174     "session" : "4c047bbc016c",
175     "input" : "/bin/busybox XUSRH",
176     "sensor" : "a927e8b28666"
177   }, {
178     "eventid" : "cowrie.command.input",
179     "timestamp" : "2017-09-17T04:06:39.859611Z",
180     "message" : "CMD: cd /dev/shm; cat .s || cp /bin/echo .s; /bin/busybox XUSRH",
181     "system" : "CowrieTelnetTransport,93,94.51.110.74",
182     "isError" : 0,
183     "src_ip" : "94.51.110.74",
184     "session" : "4c047bbc016c",
185     "input" : "cd /dev/shm; cat .s || cp /bin/echo .s; /bin/busybox XUSRH",
186     "sensor" : "a927e8b28666"
187   }, {
188     "eventid" : "cowrie.command.success",
189     "timestamp" : "2017-09-17T04:06:39.860568Z",
```

This looks familiar...

Bill-36641 Spam x

Rosa Ogram <RosaOgram@cridersvillehealthcare.com>
to conductorn

Aug 25

Why is this message in Spam? It's similar to messages that were detected by our spam filters. Learn more

Images are not displayed. Display images below

Hi,

Here is a copy of your bill.

Thank you & have a great weekend!

...

Rosa Ogram

Bill-85882 Spam x

Gretchen Jagger <GretchenJagger@pineiroproductions.com>
to als

Aug 25

Why is this message in Spam? It contains content that's typically used in spam messages. Learn more

Images are not displayed. Display images below

Hi,

Here is a copy of your bill.

Thank you & have a great weekend!

...

Gretchen Jagger

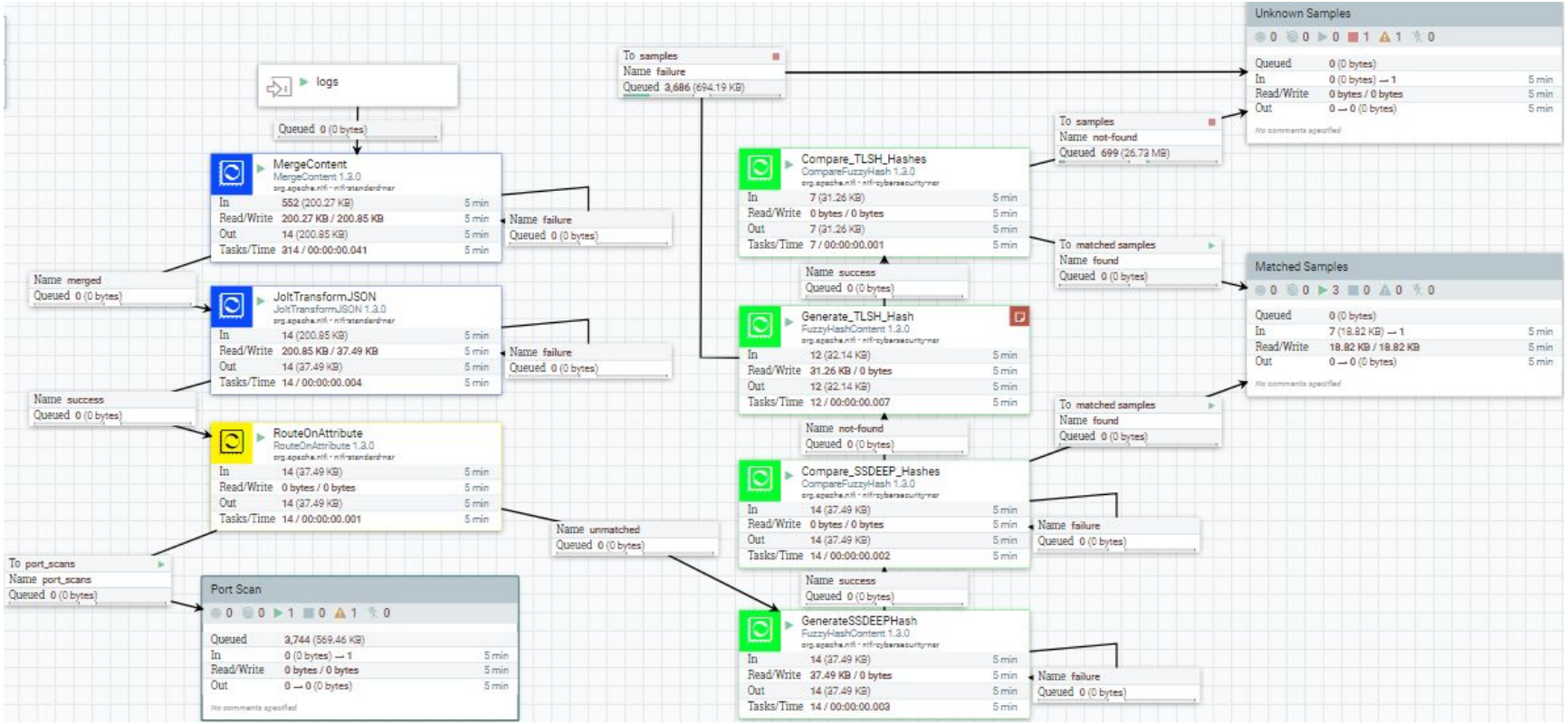
Locality-sensitive hashing

A type of algorithm that can be used to “group” similar items together and may provide a similarity score between two particular items.

Areas of application:

- Genome-wide association study
- Anti-spam (e.g. TLSH, Spamsum/SSDeep)
- Near-duplicate detection
- etc

NiFi + SpamSum + TLSH = WIN!



Wrapping up

Key points

- Treat everything as data
- Be flexible on how you build your data flows.
- Apparently unrelated domains may speed up your results
- Use MiNiFi to aggregate data at the edge whenever possible
- NiFi rocks!*

* Disclaimer: We may be a bit biased…

Future Steps

- Automate IP blocking & firewall rules (ML)
- Continuously update signature definition list with new sigs
- Analyze epidemiology & spread vectors
- Follow evolution of malware families
- Support attribution of samples

Further reading

Mysterious Hajime botnet has pwned 300,000 IoT devices
https://www.theregister.co.uk/2017/04/27/hajime_iot_botnet/

Identifying unknown files by using fuzzy hashing
https://www.honeynet.org/node/811

Classifying Malware using Import API and Fuzzy Hashing – impfuzzy
http://blog.jpcert.or.jp/2016/05/classifying-mal-a988.html

Template and samples:
https://github.com/fluenda/dataworks_summit_iot_botnet

Thank you