
# Hunting the Known Unknowns (with DNS)

Ryan Kovar and Steve Brant

Security Strategists @ Splunk

splunk>

# Tools

- **URL Toolbox**
  - https://splunkbase.splunk.com/app/2734/
- **Base64**
  - https://splunkbase.splunk.com/app/1922/
- **Common Information Model (CIM)**
  - https://splunkbase.splunk.com/app/1621/

Special thanks to Splunkers Cedric Le Roux and Sebastien Tricaud for making multiple tools that we love and adding "feature requests" whenever we come up with a new idea.

# Finding unauthorized DNS servers

- DNS Tunneling
- DNS Spoofing

Host Subnet

```
index=stream sourcetype=stream:dns dest_port=53 dest_ip!=10.0.0.0/8 | stats count by dest_ip
```

```
index=bro sourcetype=bro_dns dest_port=53 dest_ip!=10.0.0.0/8 | stats count by dest_ip
```

```
tag=dns dest_port=53 dest_ip!=10.0.0.0/8 | stats count by dest_ip
```

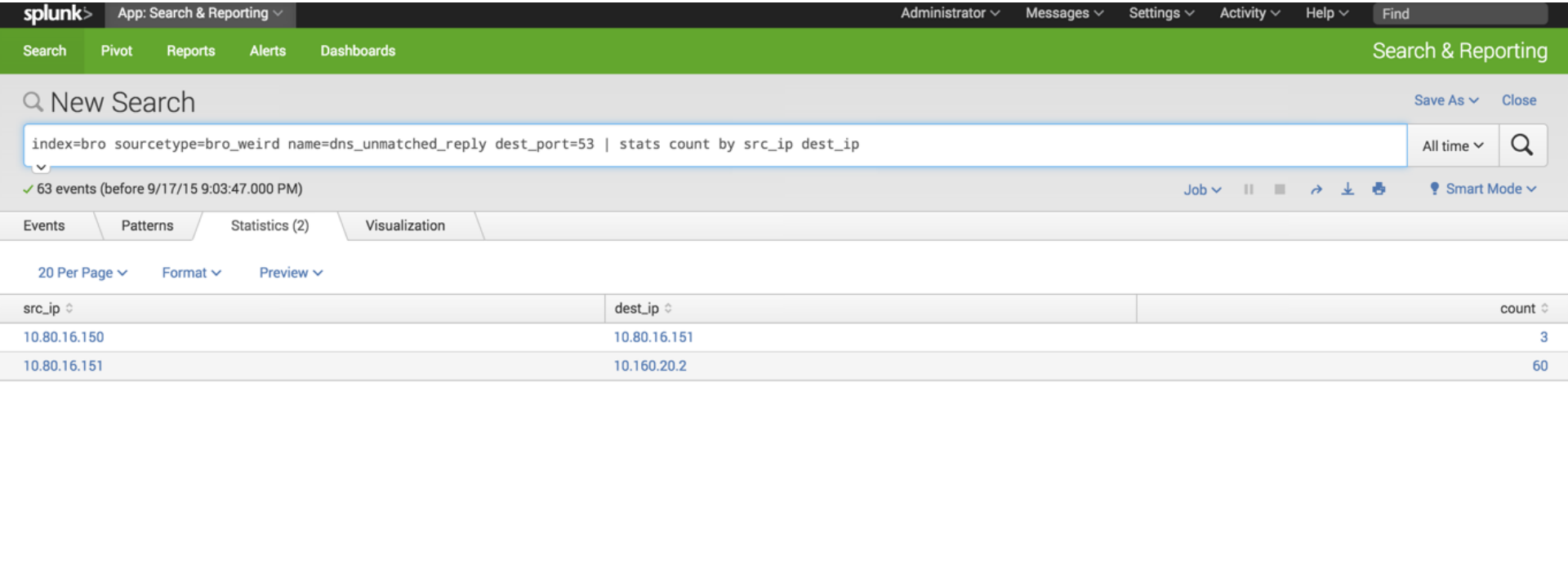
# Finding unauthorized DNS servers

# Finding Unmatched DNS Replies

- DNS Spoofing

```
index=bro sourcetype=bro_weird name=dns_unmatched_reply dest_port=53 | stats count by src_ip dest_ip
```

# Finding DNS Spoofing Activity

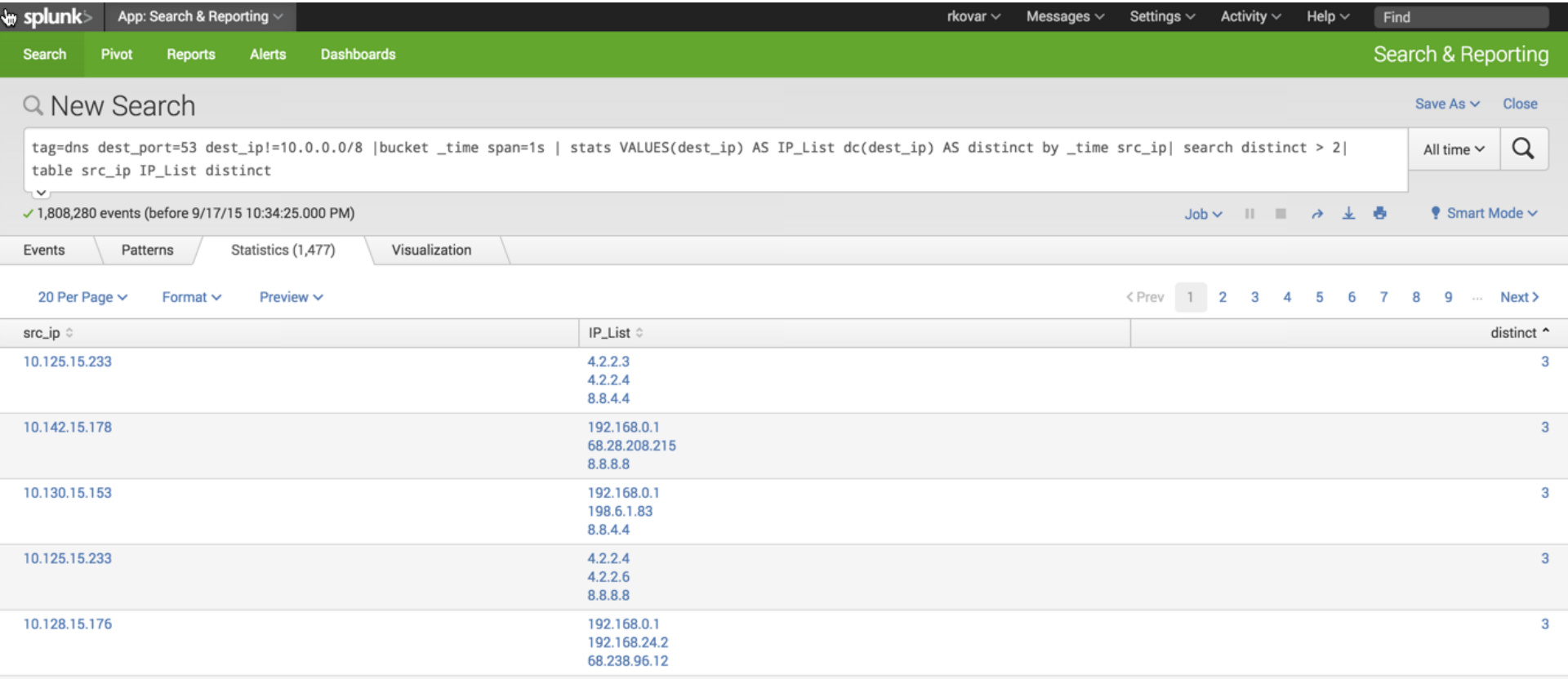


# Finding clients connecting to multiple DNS servers

- DNS Spoofing
- DNS Exfil
- DNS Tunneling

```
tag=dns dest_port=53 dest_ip!=10.0.0.0/8 |bucket _time span=1s | stats VALUES(dest_ip) AS IP_List
dc(dest_ip) AS distinct by _time src_ip| search distinct > 2|  table src_ip IP_List distinct
```

# Finding clients connecting to multiple DNS servers



# Finding Clients with extremely Looooooooooooong queries

- DNS Tunneling
- DNS Exfil

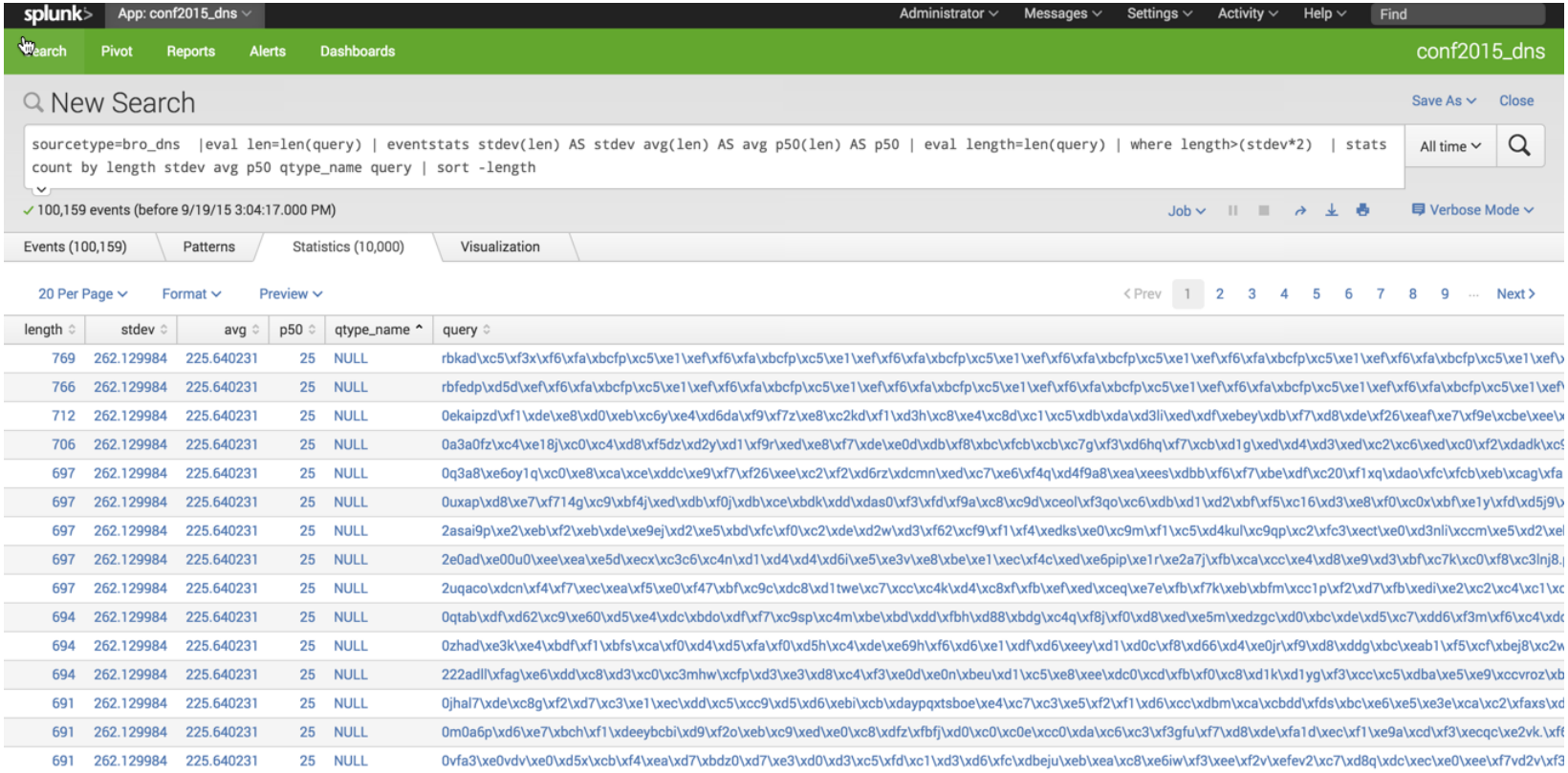
**Find anything that is 2 standard deviations**

```
sourcetype=bro_dns  |eval len=len(query) | eventstats stdev(len) AS stdev avg(len) AS avg p50(len) AS p50 | eval length=len(query) | where length>(stdev*2)  | stats count by length stdev avg p50 qtype_name query | sort -length
```

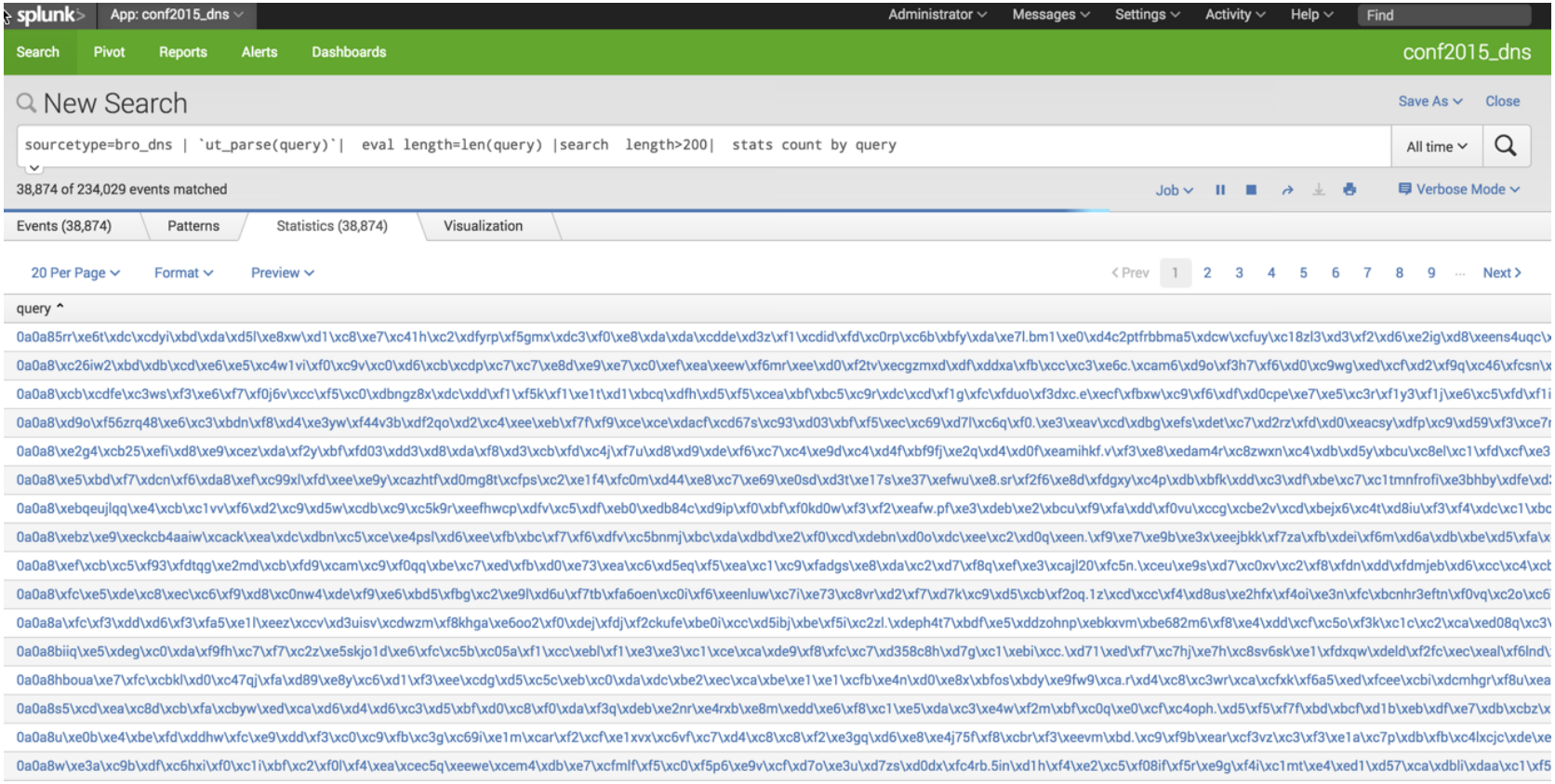
**Finding queries over 200 characters long**

```
sourcetype=bro_dns | `ut_parse(query)`|  eval length=len(query) |search  length>200|  stats count by query
```

# Finding Queries Two Standard Deviations Over Normal



# Finding Queries Over 200 characters



# Finding domains and subdomains with high Entropy

- DNS Tunneling
- DNS Exfil

**Domains**

```
sourcetype=bro_dns | `ut_parse(query)`| lookup FP_entropy_domains domain AS ut_domain | search NOT FP_entropy=* |  `ut_shannon(ut_domain)`| search ut_shannon > 4.0 | stats count by query ut_shannon
```

**Subdomains**

```
sourcetype=bro_dns | `ut_parse(query)`| lookup FP_entropy_domains domain AS ut_domain | search NOT FP_entropy=* | `ut_shannon(ut_subdomain)` | search ut_shannon > 4.5| stats count by query ut_shannon
```

# BUT FIRST…What is Entropy?

- It is the measure of randomness in a variable
  - The higher the randomness the higher the measure
- Most often “Shannon” entropy is calculated, but there are different calculations of entropy
- Example:
  - google.com
    - Shannon Entropy score of 2.6 (low)
  - A00wlkj—(-a.aslkn-C.a.2.sk.esasdfasf1111)-890209uC.4.com
    - Shannon Entropy score of 4.28 (high)

# Finding domains and subdomains with high Entropy

- DNS Tunneling
- DNS Exfil

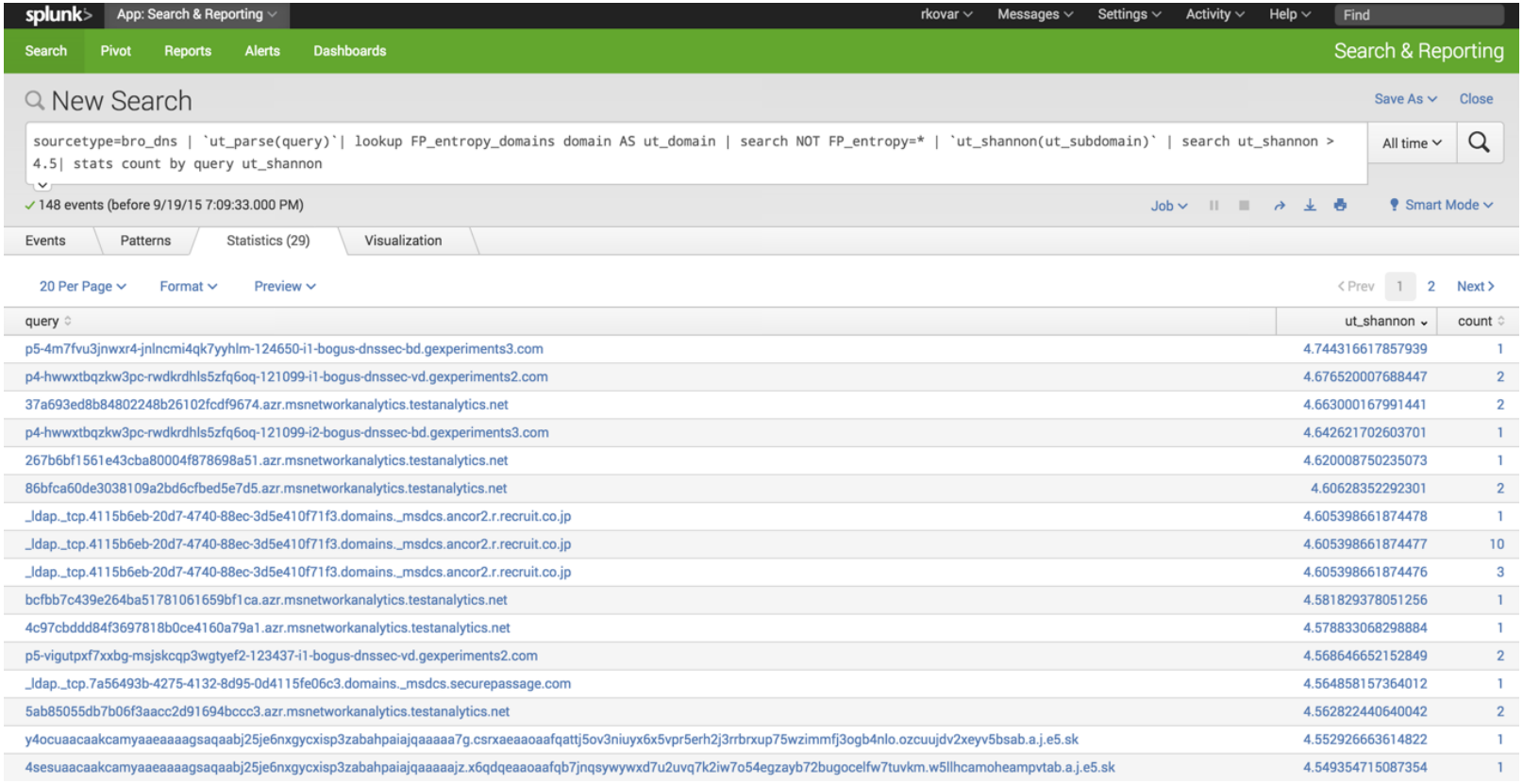
**Domains**

```
sourcetype=bro_dns | `ut_parse(query)`| lookup FP_entropy_domains domain AS ut_domain | search NOT FP_entropy=* |  `ut_shannon(ut_domain)`| search ut_shannon > 4.0 | stats count by query ut_shannon
```

**Subdomains**

```
sourcetype=bro_dns | `ut_parse(query)`| lookup FP_entropy_domains domain AS ut_domain | search NOT FP_entropy=* | `ut_shannon(ut_subdomain)` | search ut_shannon > 4.5| stats count by query ut_shannon
```

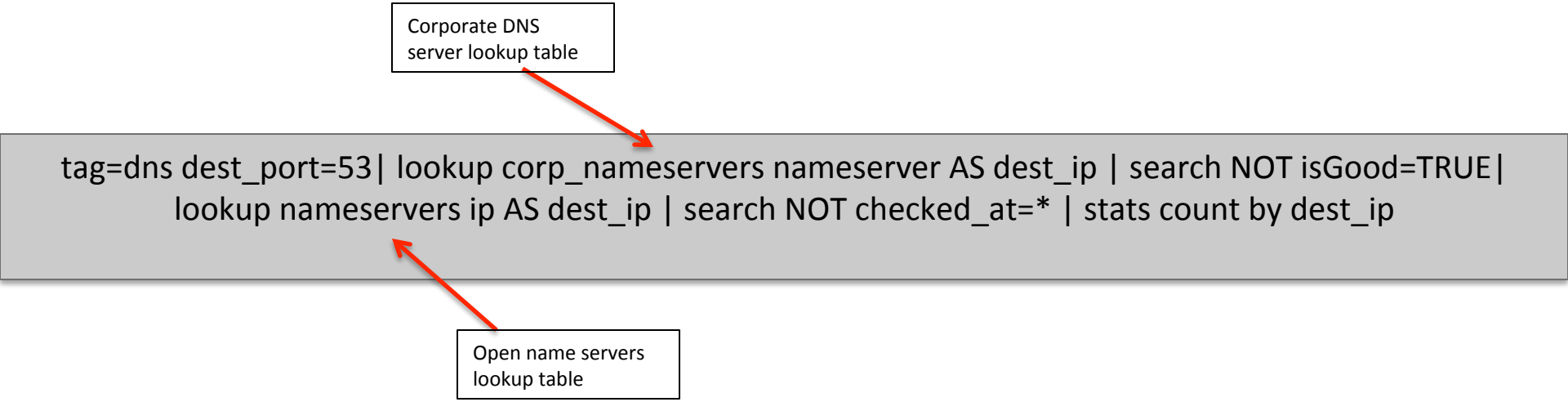
# Finding Subdomains With High Entropy



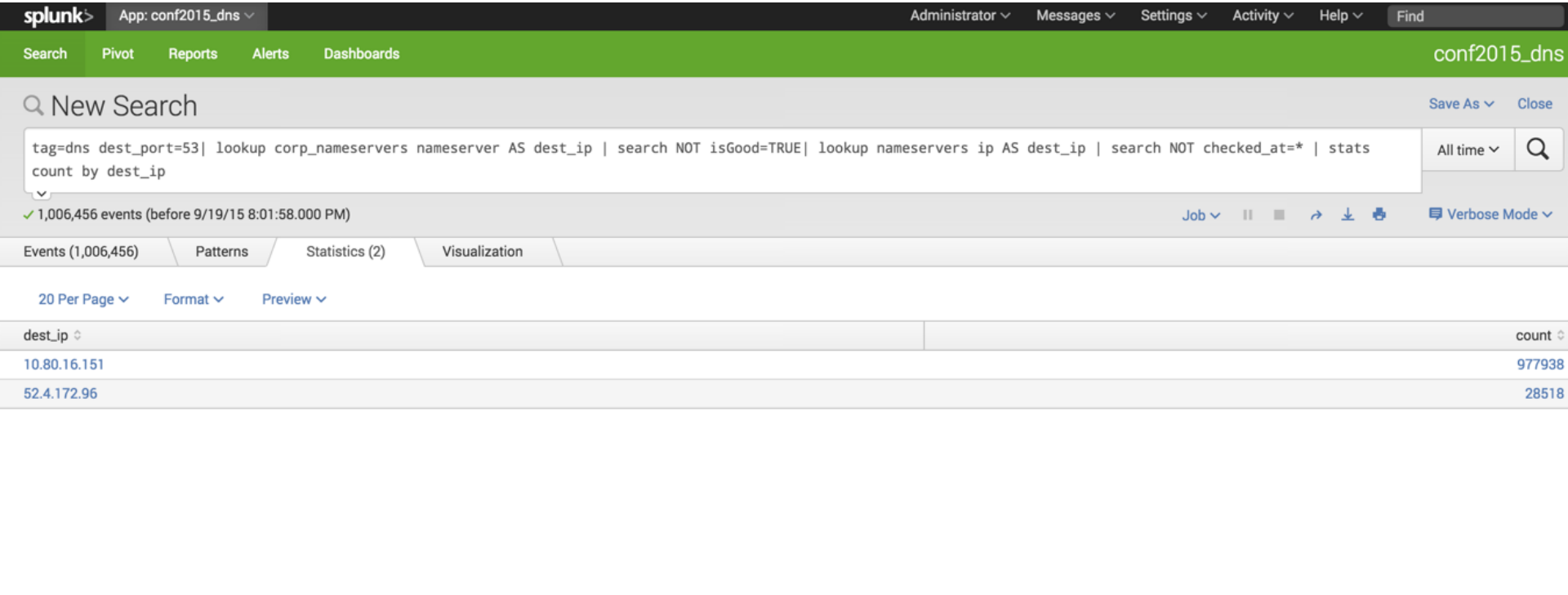

| query | ut_shannon | count |
|---|---|---|
| p5-4m7fvu3jnwxr4-jnlncmi4qk7yyhlm-124650-i1-bogus-dnssec-bd.gexperiments3.com | 4.744316617857939 | 1 |
| p4-hwwxtbqzkw3pc-rwdkrdhls5zfq6oq-121099-i1-bogus-dnssec-vd.gexperiments2.com | 4.676520007688447 | 2 |
| 37a693ed8b84802248b26102fcdf9674.azr.msnetworkanalytics.testanalytics.net | 4.663000167991441 | 2 |
| p4-hwwxtbqzkw3pc-rwdkrdhls5zfq6oq-121099-i2-bogus-dnssec-bd.gexperiments3.com | 4.642621702603701 | 1 |
| 267b6bf1561e43cba80004f878698a51.azr.msnetworkanalytics.testanalytics.net | 4.620008750235073 | 1 |
| 86bfca60de3038109a2bd6cfbed5e7d5.azr.msnetworkanalytics.testanalytics.net | 4.60628352292301 | 2 |
| _ldap._tcp.4115b6eb-20d7-4740-88ec-3d5e410f71f3.domains._msdcs.ancor2.r.recruit.co.jp | 4.605398661874478 | 1 |
| _ldap._tcp.4115b6eb-20d7-4740-88ec-3d5e410f71f3.domains._msdcs.ancor2.r.recruit.co.jp | 4.605398661874477 | 10 |
| _ldap._tcp.4115b6eb-20d7-4740-88ec-3d5e410f71f3.domains._msdcs.ancor2.r.recruit.co.jp | 4.605398661874476 | 3 |
| bcfbb7c439e264ba51781061659bf1ca.azr.msnetworkanalytics.testanalytics.net | 4.581829378051256 | 1 |
| 4c97cbddd84f3697818b0ce4160a79a1.azr.msnetworkanalytics.testanalytics.net | 4.578833068298884 | 1 |
| p5-vigutpxf7xxbg-msjskcqp3wgtyef2-123437-i1-bogus-dnssec-vd.gexperiments2.com | 4.568646652152849 | 2 |
| _ldap._tcp.7a56493b-4275-4132-8d95-0d4115fe06c3.domains._msdcs.securepassage.com | 4.564858157364012 | 1 |
| 5ab85055db7b06f3aacc2d91694bccc3.azr.msnetworkanalytics.testanalytics.net | 4.562822440640042 | 2 |
| y4ocuaacaakcamyaaeaaaagsaqaabj25je6nxgycxisp3zabahpaiajqaaaaa7g.csrxaeaaoaafqattj5ov3niuyx6x5vpr5erh2j3rrbrxup75wzimmfj3ogb4nlo.ozcuujdv2xeyv5bsab.a.j.e5.sk | 4.552926663614822 | 1 |
| 4sesuaacaakcamyaaeaaaagsaqaabj25je6nxgycxisp3zabahpaiajqaaaaajz.x6qdqeaaoaafqb7jnqsywywxd7u2uvq7k2iw7o54egzayb72bugocelfw7tuvkm.w5llhcamoheampvtab.a.j.e5.sk | 4.549354715087354 | 1 |

# Finding clients connecting to unauthorized DNS servers

- DNS Exfil
- DNS Tunneling



# Finding clients connecting to unauthorized DNS servers




BONUS ROUND

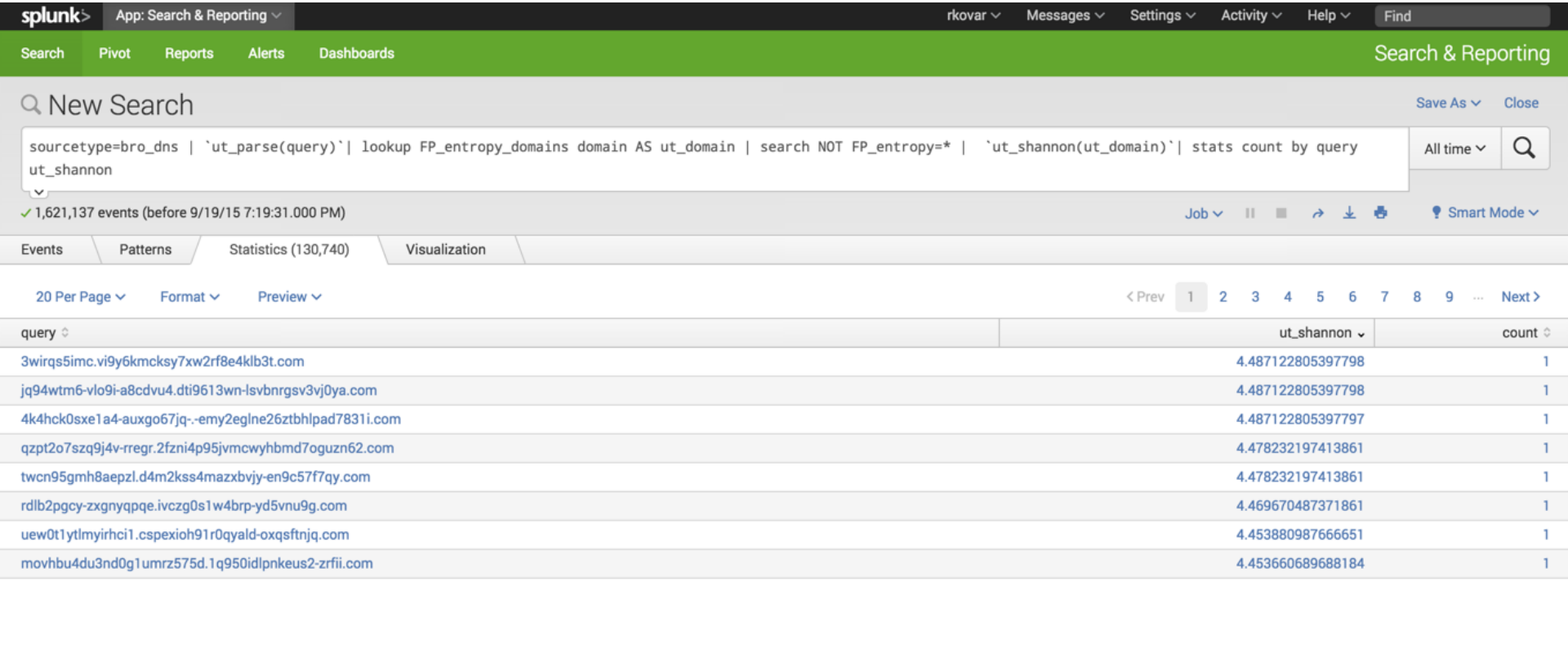
# Finding  DOMAINS With High Entropy

- Malware
- DNS Tunneling

**Domains**

```
sourcetype=bro_dns | `ut_parse(query)`| lookup FP_entropy_domains domain AS ut_domain | search NOT FP_entropy=* |  `ut_shannon(ut_domain)`| search ut_shannon > 4.0 | stats count by query ut_shannon
```

# Finding Domains With High Entropy



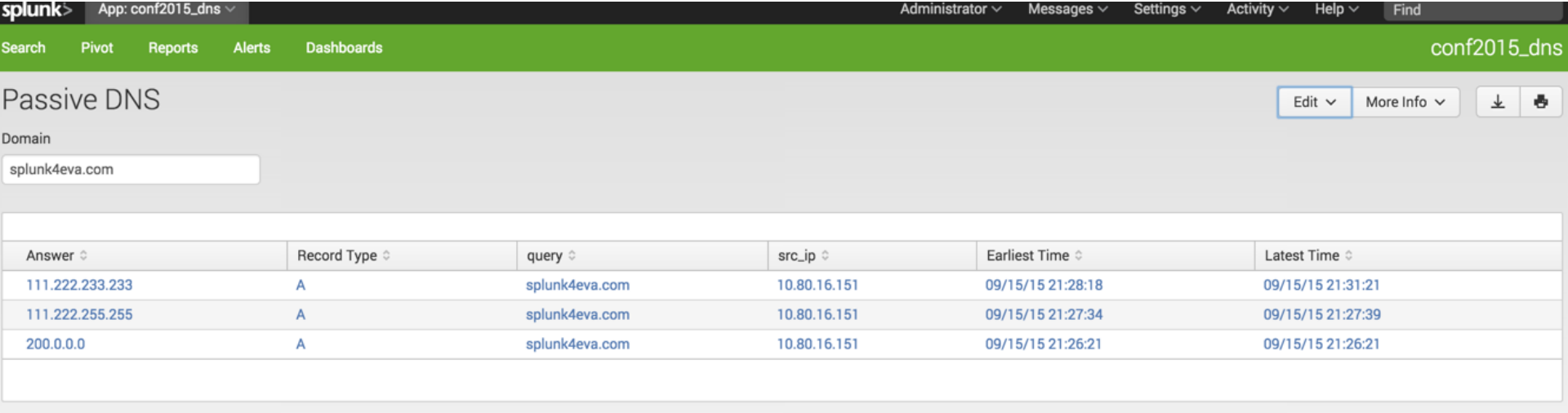
# Passive DNS dashboards

- Inspired by work done by Brian Warehime and his blog post: http://nullsecure.org/building-your-own-passivedns-feed/

**Accelerated data model version of Brian's query (requires CIM app installed)**

```
| datamodel Network_Resolution DNS search  |rename DNS.* AS * | search query=splunk4eva.mooo.com answers!="-" | stats earliest(_time) AS first_time latest(_time) AS latest_time by answers, record_type  | eval first_time=strftime(first_time, "%m/%d/%y %H:%M:%S")  | eval latest_time=strftime(latest_time, "%m/%d/%y %H:%M:%S")  | rename first_time AS "Earliest Time" latest_time AS "Latest Time" answers AS "Answer" record_type AS "Record Type"
```

# Passive DNS



# Base64 encoded DNS queries

- *NOTE* Can only detect Base64 with DNS logs that preserve case.  Bro will not work
- Further note, this uses a new tool called gather which has not yet been released but will be released shortly

**Base64 Decoded**

```
sourcetype=stream:dns  | `ut_parse(query)`|fields ut_subdomain_level_2|  base64 field=ut_subdomain_level_2 action=decode  mode=append suppress_error=True| search base64=* |  eval base64 = replace(base64, "\\\x\w{2}", " ") | dedup base64 | gather counter=2 fieldname=base64 | sort -_time | table  glue
```

**HUGE thanks to Cedric Le Roux and Sebastien Tricaud for creating great tools like FAUP, URL Toolbox, base64, and gather. Not only did they create tools we love, they implemented "Feature Requests" that we asked for within 20-30 minutes of reciept of email. Thanks guys!**

# Base64 encoded DNS queries