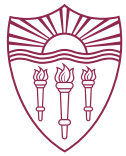

Analyzing DITL Data From B-Root At USC/ISI

Wes Hardaker <hardaker@isi.edu>

University of Southern California
Information Sciences Institute

Our Data Processing

- **Talk goal:**
  - Just another example of processing
  - (not better or worse; just different)
- **We store all B-Root DITL data locally too**
  - Example data: 2017-04-12
- **Stored as:**
  - Raw PCAPs like OARC DITL 250G
  - Also converted to text format 96G
- **Accessible via hadoop**
  - Small cluster: 34 nodes with various "cores"
- **Mappable to SQL via HIVE**

Advantages of Text Records

- **It parses ~ 3x faster**
- **Human readable**
- **Python / Awk / Perl / Emacs**
- **Other text processing tools**
  - FSDB – our format
- **Cons:**
  - We lose ICMP and other data
  - We lose bad packets
  - We lose response data sections
    - (but for the root: who cares)

Our Text Format: FSDB

- https://www.isi.edu/~johnh/SOFTWARE/FSDB/

- Basically: a tab-separated column with tools

- Example Header:

  #fsdb –F t colone coltwo colthree

DNS FSDB – output from dnsanon

- https://ant.isi.edu/software/dnsanon/
  - Converts pcap to FSDB
- **#fsdb -F t msgid time srcip srcport dstip dstport protocol id qr opcode aa tc rd ra z ad cd rcode qdcount ancount nscount arcount edns_present edns_udp_size edns_extended_rcode edns_version edns_z msglen name type class**

FSDB Tools

```
# xzcat  20170412-000108-00661294.lax.message_question.fsdb.xz    |
                dbrow '_name =~ /\.(com|net|org)\.$/'    |
                dbcol         edns_udp_size |
                dbsort -n   edns_udp_size |
                dbrowuniq –c                    |
                dbsort -n count

#fsdb -F t edns_udp_size count
643             1
815             1
…
1280            7126
4000            17804
512             40833
0               327308
4096            1012253
```

(Simplified) Python Code to Reduce Counts

```
results = {}
for line in sys.stdin:
    parts = line.split("\t")
    results[parts[0]] += 1
for tld in results:
    if results[tld] > 10000:
        print(tld + "\t" + str(results[tld]))
```

Python Code to Count TLDs

```
for line in sys.stdin:
    parts = line.split("\t")
    name = parts[28]

    labels = name.split(".")
    tld = labels[-2]

    print(tld  + "\t1")
```

Results

com 997047995
net 514338641
. 188993152
local 148892030
org 85750997
arpa 84887548
Home 60652503
\131 47566594
cn 40378241
DHCP\032HOST 37124734
home 32843087
ru 32805533
kr 32182841
localdomain 26990694
uk 25488660
lan 23623606
de 23566105
DHCP 22409704

B – how much to .com?

- hive (default)> **select count(*) as count from broot_lax_message_question_orc where subdir = '2017/04/12' and name like '%.com.';**
- Query ID = hardaker_20180309075519_9372dfd7-d70d-4962-b2bf-6fd6ed496a10
- …
- 2018-03-09 07:55:55,850 Stage-1 map = 40%, reduce = 0%, Cumulative CPU 5122.62 sec
- …
- OK
- **count**
- **993520927**
- **Cumulative CPU: 9199.53 sec**
- **Time taken: 119.897 seconds, Fetched: 1 row(s)** **(76.73x)**

HIVE: SQL over Hadoop

hive (default)> **select type, count(type) as count**
from broot_lax_message_question_orc
where subdir = '2017/04/12' and
name like '%.dns-oarc.net.' group by type;
…
OK

type	count
DS	40
AAAA	3502
A	5698
TXT	20
A6	2

Total MapReduce CPU Time Spent: **13399.49 sec**
Time taken: **151.818 seconds**
Speed-Up: **88.26x**

Conclusion

- Other formats can lead to speed-ups
- If you have a simple "is this worth studying" que
- Ask Me for a quick test?
- Ask Paul or Warren for a quick test?