P2PN-DNS

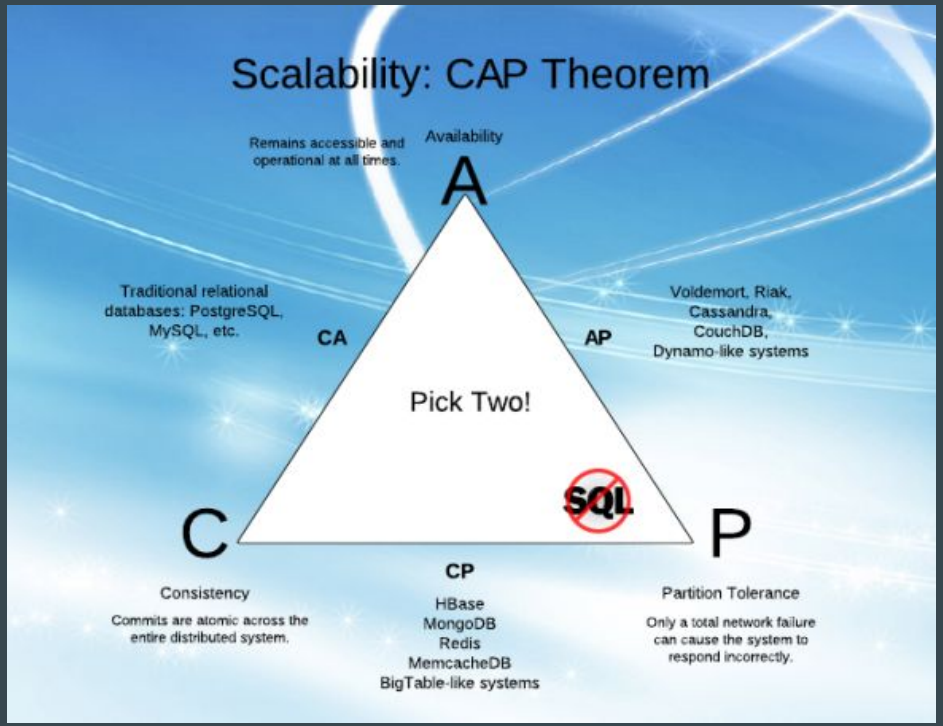
The NAT's Watch: John Grun, Arjun Ohri, Raj Balaji

Introduction

- Domain Name System (DNS) is critical element of Internet infrastructure
- Translates IP addresses to human readable names
- Thirteen root name servers in the world
- Targets for hackers, cyber terrorists, censorship, electronic warfare
- Containers would benefit from a distributed DNS solution

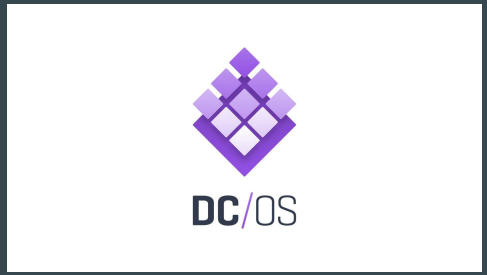
Subject and Problem

- CAP Theorem
- Automatic Peer Finding
- Trust Problem



Previous Work

- DC/OS: open source based on Apache Mesos -- Redundant DNS
- KadNode: small, unadopted, domain intercepting, DNS proxy
- Amazon: Route 53



Algorithms Used

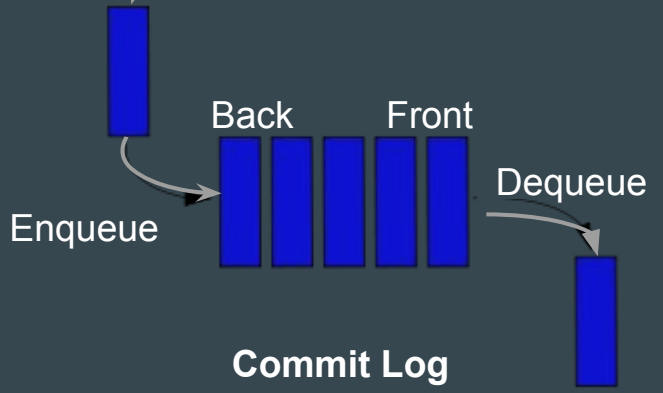
Domain Name

IP Address

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01/01/2015
K5	3,ZZZ,5623

Key Value Store

Incoming DNS update message
Key: Google.com, Value:
172.217.10.142

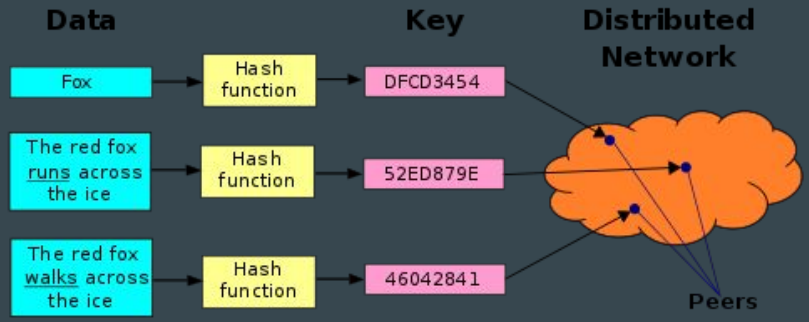


Commit Log

Write to Key, Value Store

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01/01/2015
K5	3,ZZZ,5623

Algorithms Used



Distributed hash table (DHT)

Contribution

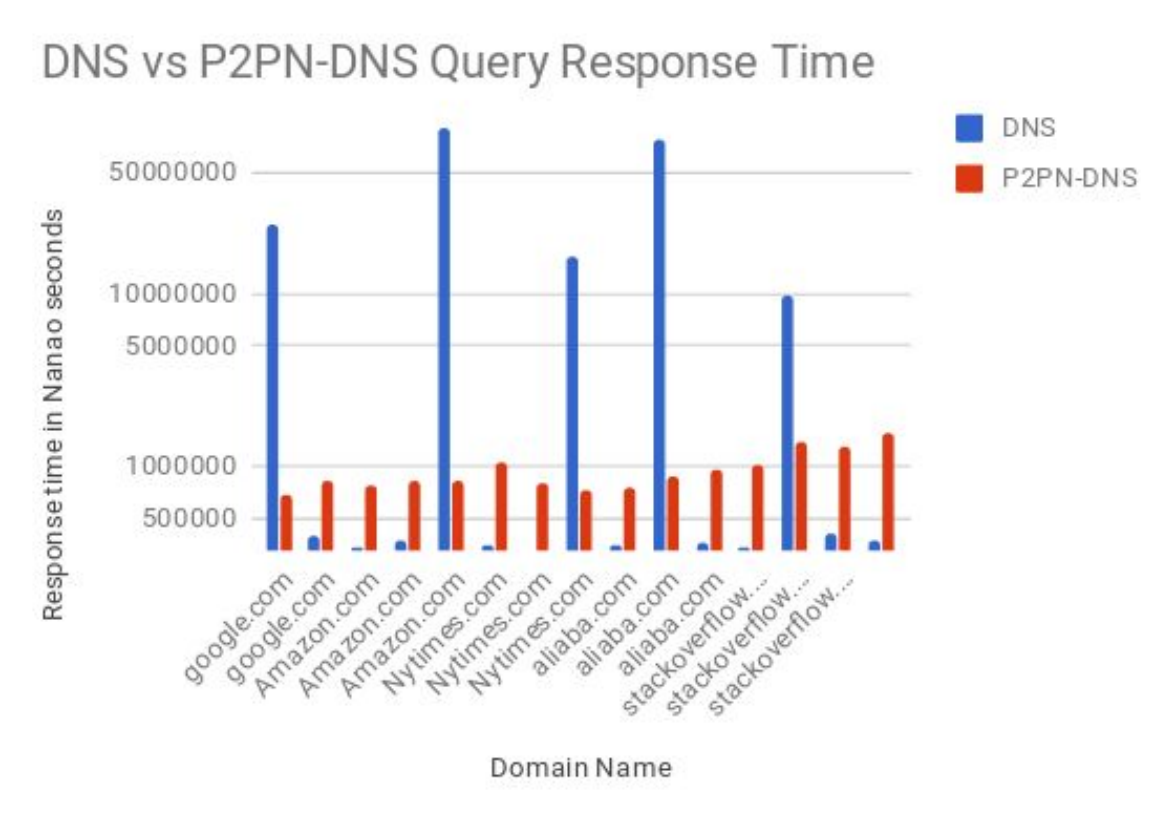
- Creation of our own Peer to Peer Domain Name System
- Syncing data between nodes
- Ensuring consistency, availability, and partition tolerance
- Focusing on functionality and technicality over political agenda
- Distributed DNS solution that can be deployed for containers

Methodology

- Response time to DNS queries
  - DNS query to P2PN-DNS and dnsmasq
  - Compare round trip for query and response
- Availability - Loss of Nodes
  - 1.) Start with multiple nodes (3) and send DNS update to one node.
  - 2.) Query nodes for DNS record.
  - 3.) Remove one node. Repeat 1-3 until one node remaining
  - 4.) Log failures to retrieve DNS record
- Average time to sync records across nodes
  - Enable logging
  - Start multiple nodes
  - Send Dns update to one node
  - Observe amount of time it takes for node update to appear in logs

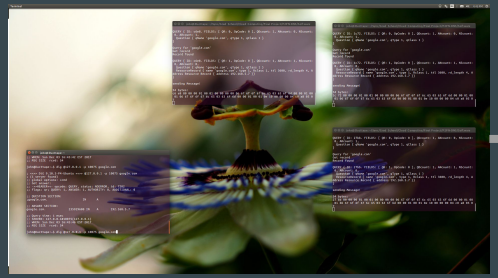
Results: Response time to DNS queries

Comparison of DNS query times between P2PN-DNS and standard DNS(dnsmasq)

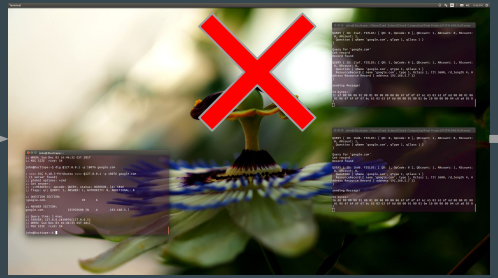


Results: Availability - Loss of Nodes

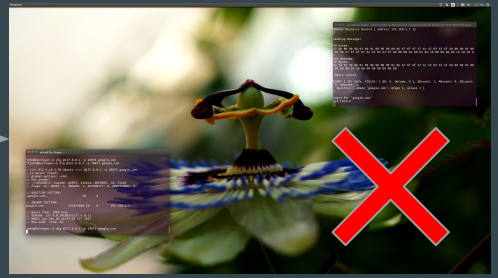
Three nodes



Two nodes



One node



Results: Average time to sync records across nodes

```
/var/log/syslog | grep P2P
P2PN-DNS50666: put: adding baea954b95731c68ae6e45bd1e252eb4560cdc45 -> Value[id:79aabc5300043139 Data (type: 0 ): c0b24e05]
P2PN-DNS50666: [store baea954b95731c68ae6e45bd1e252eb4560cdc45] changed
P2PN-DNS50666: [store baea954b95731c68ae6e45bd1e252eb4560cdc45] 0 remote listeners
P2PN-DNS50666: [search baea954b95731c68ae6e45bd1e252eb4560cdc45] [node a56f109aaa4c6451e39d9677524bdcf20eb6686c 127.0.0.1:50668] sending Query[SELECT id,seq ]
P2PN-DNS50666: [search baea954b95731c68ae6e45bd1e252eb4560cdc45 IPv6] expired
P2PN-DNS50666: Announce done IPv6 0
P2PN-DNS50668: [node 00bfe70f6bf444f69bc19afb931fb183fa70735d 127.0.0.1:50666] got 'get' request for baea954b95731c68ae6e45bd1e252eb4560cdc45
P2PN-DNS50668: [node 00bfe70f6bf444f69bc19afb931fb183fa70735d 127.0.0.1:50666] sending 1 values
P2PN-DNS50666: [search baea954b95731c68ae6e45bd1e252eb4560cdc45] [node a56f109aaa4c6451e39d9677524bdcf20eb6686c 127.0.0.1:50668] sending 'put' (vid: 274745)
P2PN-DNS50666: sending 41 bytes of values
P2PN-DNS50668: [node 00bfe70f6bf444f69bc19afb931fb183fa70735d 127.0.0.1:50666] got 'put' request for baea954b95731c68ae6e45bd1e252eb4560cdc45
P2PN-DNS50668: [store baea954b95731c68ae6e45bd1e252eb4560cdc45] storing Value[id:79aabc5300043139 Data (type: 0 ): c0b24e05]
P2PN-DNS50668: [store baea954b95731c68ae6e45bd1e252eb4560cdc45] changed
P2PN-DNS50668: [store baea954b95731c68ae6e45bd1e252eb4560cdc45] 0 remote listeners
P2PN-DNS50666: [search baea954b95731c68ae6e45bd1e252eb4560cdc45] [node a56f109aaa4c6451e39d9677524bdcf20eb6686c 127.0.0.1:50668] got reply to put!
P2PN-DNS50666: Announce done IPv4 1
```

- Syslog output from two nodes (P2PN-DNS50666 and P2PN-DNS50668)
- Dns update packet sent to P2PN-DNS50666
- Consistency between nodes obtained in under 1 second
    - Dependent upon network

Future Prospects

- Support the rest of the DNS protocol
  - CNAME, AAAA, TXT, etc records
  - Implement the true update message (DDNS)
  - Implement authentication to update DNS records
- Automatic detection of other P2PN-DNS nodes on network
- Authentication between P2PN-DNS nodes
- Hash based record ordering scheme
  - Current version uses timestamps for record ordering.
  - Merkle Tree -- Hash tree or Blockchain(without proof of work)
- Test partition tolerance

Conclusions

- Efficient response time to DNS queries
- Formulaically sustaining CAP theorem
- Distributed DNS solution
- Better Security than normal DNS

Demo (time permitting)

- Start multiple nodes
- Send DNS update to one node
- Send DNS query to another node
- IP address returned from query should match the update