A Survey of Analysing the Internet DNSSEC traffic for resolving platform optimisation and performance improvement

R Venkateswara Rao
M.Tech student, Computer Network Engineering,
REVA Institute of Technology and management,
Bengaluru.

Mr. G C Sathish
Senior Associate Professor, Dept of CSE
REVA Institute of Technology and management,
Bengaluru

Abstract: In the Internet of Thing (IOT) scenario and the expanding internet users very fast with accelerating scalability of infrastructure, the DNSSEC enabled resolvers also expanding correspondingly. It was already conformed by measurements that DNSSEC resolving needs 4 to 5 times more platform compared to Plain DNS system. And there were some measurements researches have carried out to resolve the user-friendly notation to machine-friendly notation (name to IP resolution and vice versa). Plat form optimisation and efficient way of resolving the DNSSEC traffic. We want to analyse the traffic with R language by using optimum principal component analysis (PCA) and K-means cluster analysis with mixed integer linear programming for routing as unsupervised machine learning.

Keywords: DNS, DNSSEC, PCA, PCAP, unsupervised machine learning, k-means clustering, R packages and tools.

I. Introduction

Beginning communication between two devices used to be direct connection and for multiple devices network based setup practiced and to have communication devices used to be numbered to get identified and used for resource sharing. As devices in network grew, Domain Name System (DNS) was adopted. In order to remove conflict in naming a fully qualified domain name (FQDN) system was proposed. As DNS protocol is widely used by users and ISPs (Internet service provider) network administrators can not block all DNS traffic, which leads to the Cyber attacks like DDOS, cache poisoning, DNS Amplification attack, DNS Tunnel based botnet communication, etc.

DNSSEC provides Integrity (DNS response was not changed during its transmission) and authenticity (the DNS response really sent from where it claims from) of DNS response in name resolution.

At the same time it has increased the overhead for resolving and validating user-friendly (name) to machine-friendly (IP) and vice versa. We present our study as given below

II. Literature survey

In [1, 8] the authors explained the DNSSEC traffic resolution using R packages by considering network related indicators and DNS-related indicators. It was experimentally demonstrated by authors that the resolving resources reduced to 3.5 instead of 5 times by using Distributed Hash Table (DHT) with proactive caching mechanism and traffic split between nodes based on FQDN instead of IP addresses. The same authors performed experiments and found that DNSSEC resolution depends on the implementation and it ranges between 169% and 500% more resources.

In [2] the authors proposed the novel architecture, a new DNSSEC validation mechanism that share the load between the resolving servers and clients as clients also DNSSEC enabled to some extent to do some validate processing at their end and reduced the load on resolving servers without compromising the integrity and authenticity and showed high performance.

In [3], the authors demonstrated that the low negative caching mechanism in DNSSEC is vulnerable to DoS/DDoS attacks and proposed a high efficient Negative Caching for DNSSEC-Oblivious resolvers (NCDO). NCDO uses NSEC/NSEC3 RRs to cover name span instead of single name. Its functions are enhancing cache hit rate, response time and cache consistency.

In [4], authors could measure the validating caching resolvers and non-validating caching resolvers using DSR; DS ratio

In [5], authors studied and proposed the check repeat, a new query-based method to find out the DNSSEC resolvers as validation is important before resolving.

In [6], authors proposed and investigated CPU usage of the DNSSEC resolving servers. in this they proposed the novel architecture which they call as PREFETCHx which reduces required nodes to 4 times less. Current resolving servers use IPxor and IP address based resolution whereas authors used FQDN's popularity Distribution (Zipf).

In [7] the author proposes the adaptive k-means cluster algorithm where in the selection of the initial element is not dependent of cluster presentation. Usually the data set is partitioned into k clusters but identification of cluster is again dependent on the initial selection. Whereas author fixed the clusters but adaptively merging or creating the new one depending on the close prosperities of the given element.

III. Proposed system

DNSSEC is the security extension of DNS and it needs five times more resolving resources [1, 8]. In order to improve performance and resource optimisation, FQDN resolutions cost should be minimised and to minimise it we need to consider the metrics

a) Network costs (servers occupation time)
b) Computation costs (signature checks)
c) Memory costs (cache related activities)

Literature indicates that lot of research done towards secured FQDN resolution but there are still open issues in Secure Domain Name System Deployment and it is very difficult to make out which zone is DNSSEC-capable.in reality, all zones are not DNSSEC-capable and it is evolving. If it comes to mobile devices as it moves different zones which may be DNSSEC- capable or not, resolution must be done.

Introduction to R language:

R is a scripting programming language and software environment for statistical analysis, graphics representation and reporting. R Basic Data structures: Vectors, Matrices, Arrays, Lists, Data frames etc,. R can interface with most of the format files like CSV, Excel etc, generates graphs: box plot, scatter plot, bar graph etc, R is being used by statistical analytics and Data mining applications.

Using R for analysing DNSSEC traffic:

Step1: collection of the pieces of packet capturing (PCAP) DNS data. Raw data collected from cache

Step2: Cleaning the data and pre-process it(Fig 1)

Step3: save the data as data frame R language compatible format.

Step4: use the statistical and probabilistic methods to further study most influential parameter.

Step5: applying the principal component analysis on data set to reduce the dimension.(Fig2)

Step6: applying the K-means clustering to narrow down the data to centralised component. (Fig 3&4)

Step7: use is made to apply boxplot to find out the most influence parameter and many R tools like box plot to study outliers and their influence.

Step8: Finally adopting most suitable algorithm to achieve good efficient resolving mechanism. This analysis with R not only optimises platform like distributing the load between different resolvers and it also improves performance as CPU related memory operations with reduced parameters carried out.

Literature survey indicate that plat from optimisation is possible when compared to conventional DNS and DNSSEC architectures and systematically analysing the principal components.Table1 gives the platform optimisation invented by different researchers.

Table1: platform optimisation chart

Architecture	Name system	Resource required
IPxor	DNS	X
IPxor	DNSSEC	5X
PREFETCHx	DNS	0.85X
PREFETCHx	DNSSEC	3.5X

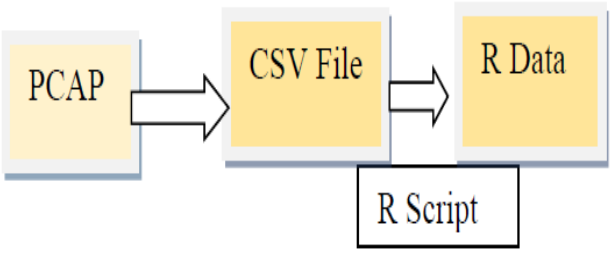

Fig 1: shows how PCAP data is ported to R data frame

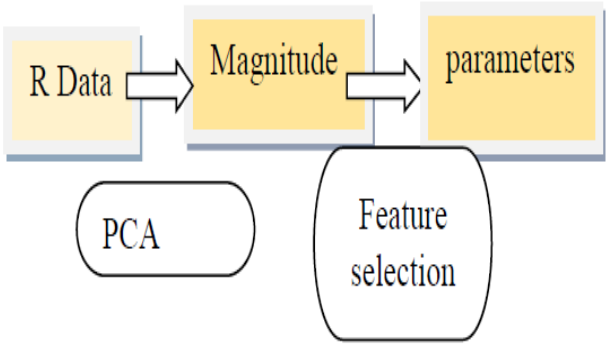

Fig 2: Reducing Dimension via PCA

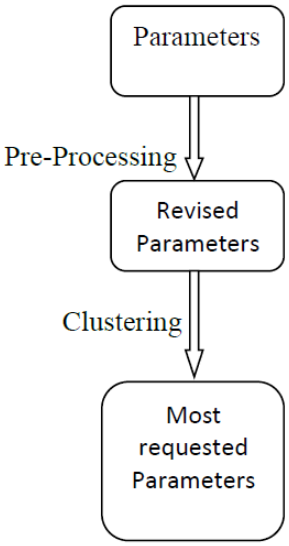

Fig 3: After feature selection, pre-processing and clustering

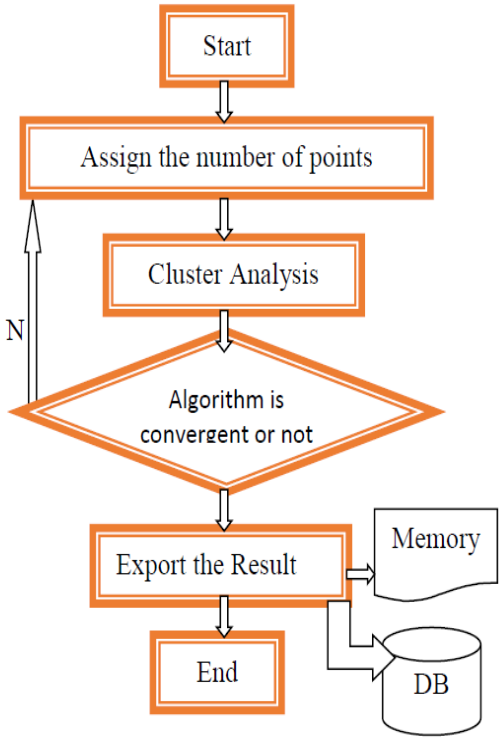

Fig 4: K-Means Clusters algorithm flow chart

IV. Conclusion

As DNSSEC resolvers and valuators need many resource records in order to have authenticity, data integrated and authorisation. Some RRs related to Network. Some may be related to computation and some others relates to memory related. And DNSSEC involves huge data query/Reponses and Trust Anchor Repositories (TARs). Analysing big data with analytical and probabilistic tools and selecting effective load balancing and distributing algorithm we can optimise the resource and performance can be improved. It is very difficult to find out recursive DNS resolvers in the present scenario and there is very little work was done as access to systems (resolving or recursive servers and authoritative servers) is difficult. Only ISP providers and owners can have access to the resolvers. With limited data set analysis of platform optimisation carried out using R tools and unsupervised machine learning technique and it gives satisfactory results.

V. Acknowledgment

I thank my internal guide G C Sathish, Sr. associate professor for continuous guidance and principal, Dr Sunil Kumar Manvi for helpful tips; encouragement, college for providing the facility and finally I thank my colleagues and friends for helping me out while discussing the issues.

VI. References:

[1] Yanchang Zhao and Yonghua Cen,Text book "Data mining and applications with R ",(2015 edition)Chapter -15, pages 435- 455.

[2] YongJin, Masahiko Tomoishi and NariyoshiYamai "An Advanced Client Based DNSSEC Validation and Preliminary Evaluations toward Realization" (IEEE 2016)

[3] ZhengWang,"Optimizing Negative Caching for DNSSEC-Oblivious Resolvers" 2015 IEEE 14th International Symposium on Network Computing and Applications

[4] Kensuke Fukuda, Shinta Sato and Takeshi Mitamura "A Technique for Counting DNSSEC Validators" 2013 Proceedings IEEE INFOCOM

[5] Yingdi Yu, Duane Wessels, Matt Larson and LixiaZhang "Check-Repeat: A New Method of Measuring DNSSEC Validating Resolvers" The 5thIEEEinternational traffic monitoring and analysis Workshop(TMA 2013)

[6] Daniel Migault, St´ephaneS´en´ecal, StanislasFrancfort, Emmanuel Herbert and Maryline Laurent "PREFETCHing to overcome DNSSEC Performance Issue on large Resolving Platform" 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications

[7] Sanjiv K. Bhatia"Adaptive K-Means Clustering" American Association for Artificial Intelligence(www.aaai.org) - 2004

[8] Daniel Migault, St´ephaneS´en´ecal, StanislasFrancfort, Emmanuel Herbert and Maryline Laurent "Overcoming DNSSEC Performance Issues with DHT-based Architectures" 2013 IFIP.