Implementing encrypted DNS in Fedora and Kubernetes Clusters with FreeIPA DNS

Josep Andreu Font

Ramon Gordillo

FreeIPA
Open Source Identity Management Solution

FOSDEM'26

Who we are: Ramon Gordillo

I am Principal Solution Architect @ Red Hat

Passionate about technology, convinced on Open Source

Recurring speaker on events and several meetup groups

Member of several communities (Kubernetes community, Eclipse, etc).

https://www.linkedin.com/in/ramongordillo/
https://github.com/rgordill

FreeIPA
Open Source Identity Management Solution

FOSDEM'26

Who we are: Josep Andreu Font

I am Senior Technical Account Manager @ Red Hat

Proud Red Hat Certified Architect Level XIV, CKA, CKAD, CKS

I'd love to to talk in events, specially FOSDEM

Apart from my work @ Red Hat, I'd teach classes in University: Cybersecurity Master's Degree and final year of Computer Science Degree

FreeIPA Open Source Identity Management Solution

FOSDEM'26

The origins & motivations

Why DNS traffic needs to be secure

- Unencrypted DNS traffic can be susceptible to **eavesdropping** (inspection) by attackers on the network path, which compromises user privacy and reveals browsing history.
- Plain DNS queries are vulnerable to **on-path tampering** (manipulation), where attackers can intercept and modify the response to redirect users to malicious websites.
- The transition to encrypted protocols like DNS over TLS (DoT) is essential for a zero-trust architecture, mandating that traffic be authenticated, authorized, and encrypted to protect against these threats.

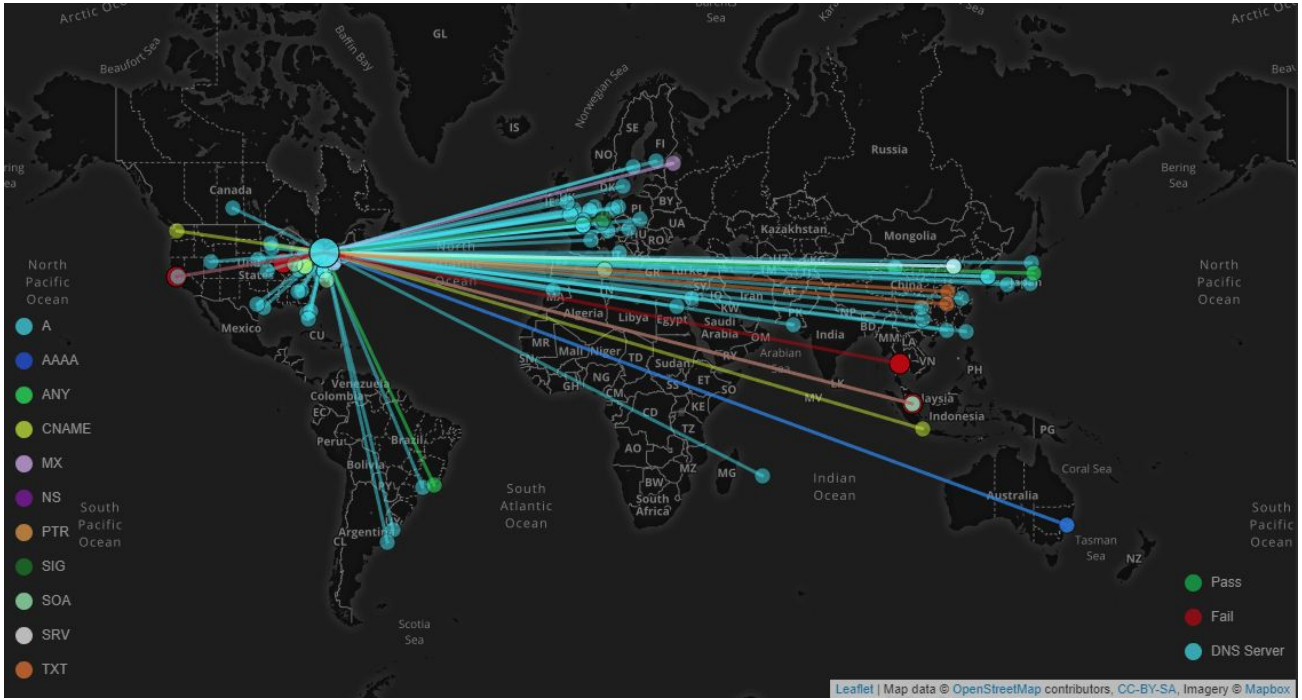
Source: https://datatracker.ietf.org/doc/html/rfc7858

Why DNS traffic needs to be secure

- Driving force is the U.S. government memorandum “Moving the U.S. Government Toward Zero Trust Cybersecurity Principles (MS-22-09)”, which mandates that internal networks supporting **hybrid workloads** must adhere to **zero-trust architecture (ZTA)** principles.
- In the EU we have the Cyber Resilience Act which is a legal framework that describes the “cybersecurity requirements for hardware and software products with digital elements”.
- To secure the communication we can use the FreeIPA Dogtag provided certificates or your own certificates.
- The encryption is provided not only on runtime, but also on boot and installation time.

Source: https://datatracker.ietf.org/doc/html/rfc7858

An example



Source: https://github.com/cchesley2397/DNS-Map/blob/master/visualizer/demo.PNG

The transition

DNS over TLS (DoT)

- DNS over TLS is an encrypted network protocol defined by IETF RFC 7858.
- TLS handshake establishes an encrypted channel before any DNS message is exchanged.
- DNS messages remain unchanged (same format as classic DNS) but are sent inside the TLS tunnel, port 853.
- TCP is required, unlike traditional DNS which typically uses UDP.
- Certificate validation ensures the resolver is legit and prevents MITM attacks.

Alternatives

- DNS over HTTPS (DoH)
  - Tunnel DNS over HTTPS. Useful if port 853 is blocked and 443 is allowed (common)
  - Hard to block, browser-native
  - More complex stack, centralization concerns
- DNS over QUIC (DoQ)
  - Encrypted QUIC over UDP, reducing overhead (negotiation, control, etc.)
  - Low latency, mobile-friendly
  - Limited support today

The question

What is the performance impact of the change?

The procedure

Setup

- Relative small instances:
  - 4 vCPUs, 8 GB RAM (AWS c8i.xlarge) for VMs
  - 8 vCPUs, 16 GB RAM (AWS c8i.2xlarge) for Kubernetes
- Performance client: **dnsperf** from DNS-OARC
- Monitoring (prometheus) and dashboards (grafana)
  - Measuring throughput (QPS) and latencies (orders of ms) from client
  - Checking resource consumption and looking for bottlenecks, and tune according to that
  - **Note**: we did previous synthetic network tests to ensure we were not hitting any limit

https://www.dns-oarc.net/tools/dnsperf

Procedure

- Create a huge amount of records in DNS zone ~1000 records with TTL 1 second.
- Configure FreeIPA as the BIND server of a K8s cluster.
- Modify params to tune FreeIPA to achieve its limits, CPUs ~100% usage, orders of ~100k QPS, logs, etc.
- Modify dnsperf to stress FreeIPA server, 200 concurrent clients with 1000 outstanding queries during 120 seconds.
- Perform the same tests with traffic unencrypted (UDP/TCP) vs encrypted (DoT). Check the difference between them.
- Load, observe and interpret the data obtained in Grafana.

https://www.dns-oarc.net/tools/dnsperf

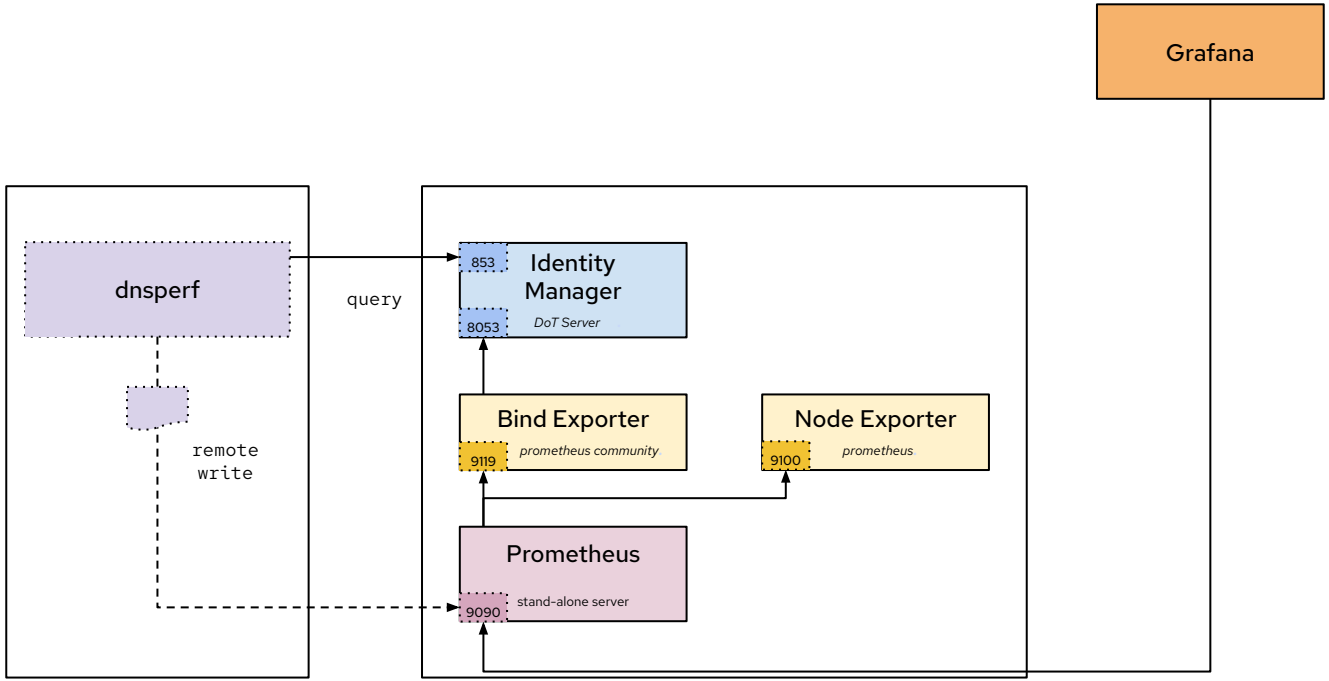
Measurements

- QPS (Queries per Second): We have measured the throughput obtained from the unencrypted (UDP/TCP) vs encrypted (DoT) tests. We see initial burst, stabilization and lastly sustained throughput until the end of the window.
- p50 (median latency): Represents the time within which 50% of queries complete.
- p99 (tail latency): Where 1% queries take longer than the time reported in this measure.

https://www.dns-oarc.net/tools/dnsperf

The architecture & results

Host to Host


Grafana
dnsperf
query
853
Identity Manager
8053
DoT Server
remote write
Bind Exporter
9119
prometheus community
Node Exporter
9100
prometheus
Prometheus
9090
stand-alone server

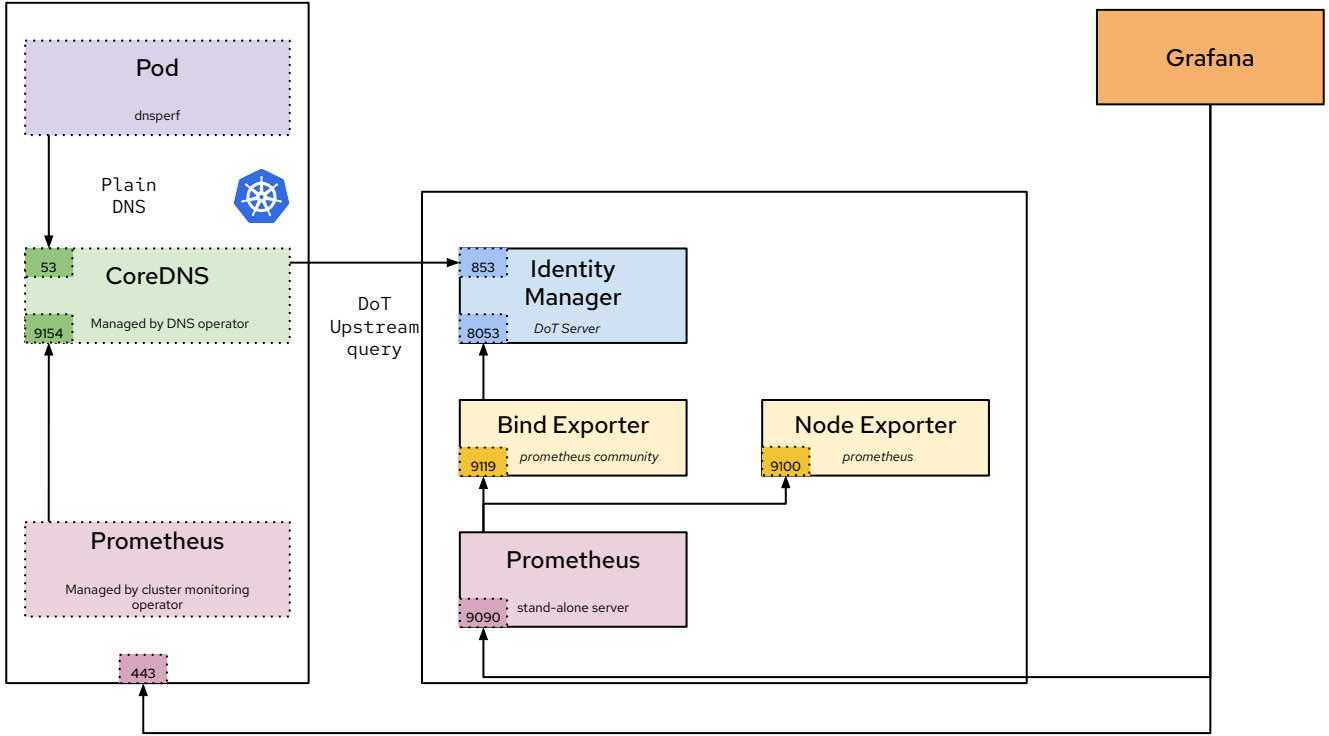
Host to Host

Metric	UDP	TCP	DNS over TLS (DoT)
Queries per Second (QPS)	304-306k	250-255k	190-192k
Query Average Latency (p50)	~970 µs	~3 ms	~3.6-3.8 ms
Tail Latency	2.8-7.2 ms	12.5-14.2 ms	18-20 ms

Host to Host

- UDP → TCP (Reliability Cost):
  - Drop ~17% (53k QPS) -> Cost of ensuring packet delivery and reliability
  - UDP p50 extremely fast, microseconds response
  - Latency triples 3.0 ms, the physics of TCP requires a full round-trip handshake before data flows of roughly 3ms.
- TCP → DoT (Encryption Cost):
  - Drop ~23% (61k QPS) -> Overhead for cryptographic operations
- Total Capacity Impact (UDP → DoT):
  - Drop ~37% (114k QPS)
  - Latency increases another 23% over TCP, which in absolute terms is relatively small ~0.8 ms
- The system is highly capable. Sustaining 190k QPS over encrypted DoT is an impressive result, proving the infrastructure handles the "crypto tax" well without collapsing.

Kubernetes to Host

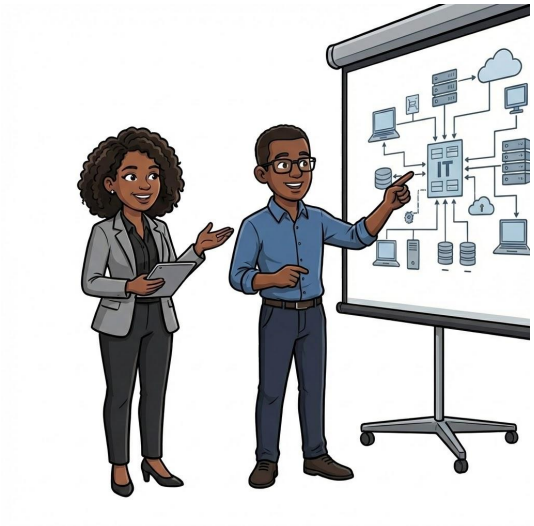


Kubernetes to Host

Metric	Pure UDP	Hybrid DoT
Queries per Second (QPS)	117-119k	117-119k
Query Average Latency (p50)	~2.2 ms	~2.37 ms
Tail Latency	9.29 ms	13.1 ms

Kubernetes to Host

- Both configurations have identical sustained throughput.
- Enabling TLS encryption on the upstream leg incurs no performance penalty. p50 latency increased marginally (0.17 ms).
- FreeIPA scales extremely well; CoreDNS is the limiting factor and as such is outside of the FreeIPA BIND domain.
- It delivers the same throughput as the insecure baseline with a statistically negligible latency impact for 99% of traffic. The security gain (encrypted credentials/queries) vastly outweighs the 4ms increase in tail latency.



Demo

References

https://www.rfc-editor.org/rfc/rfc7858

https://www.dns-oarc.net/tools/dnsperf

https://github.com/prometheus-community/bind_exporter

https://github.com/prometheus/node_exporter

https://coredns.io/plugins/forward/

https://github.com/rgordill/dnsperf-prometheus-stats


Thank You
Vinaka
köszönöm
Kiitos
Maake
Asante
Shukria
Dhanyavadagalu
감사합니다
Kam Sah Hammida
Manana
Dankon
Matondo
Dank Je
Danksheen
спасибо
Mauruuru
Biyan
Kasih
Terima
Taiku
Tack
Blagodaram
Ngiyabonga
Dziekuje
Chokrane
Diolch i Chi
Juspaxar
Arigato
Grazie
Gracias
Mochchakkeram
Bedankt
Dàkujem
Tingki
நன்றி
ধন্যবাদ
cảm ơn bạn
Ua Tsaug Rau Koj
Grazas
Gratias
Tibi
Paldies
Khun Khap
Obrigado
Děkuji
Nirringrazzjak
Di Ou Mèsi
Kia Ora
Kop
ありがとう
Suksama
Hvala
Djiere Dieuf
Rahmat
Nuwun
Matur
谢
Welalin
Misaotra
Xbana
Danke
Merci
Go Raibh Maith Agat
Eskerrik Asko
Salamat
ขอขอบคุณคุณ
Najis Tuke