DNS in a 5G Network Core: New Requirements and Demands

1. Executive Summary

Initial 5G deployments are already underway, and planning for a 5G future is part of the strategy for all mobile operators over the next few years. However, the role of DNS in fulfilling the vision of 5G has not been explored to any great depth.

This whitepaper explains the key components of the 5G vision, and shows how legacy DNS architectures are not sufficient to meet the 5G requirements in areas such as latency, security, edge computing, and IOT. Also it demonstrates how 5G requirements such as orchestration and network slicing create new requirements for DNS software and architecture.

Finally the capabilities of the PowerDNS software in meeting the requirements of a 5G network are explored.

2. The Vision for 5G Services

5G is more than an update to the Radio Access Network over previous iterations such as LTE; it is a completely new vision and architecture for mobile networks, incorporating not just improved bandwidth, but wholesale architectural changes towards an SDN/NFV based network, and specific goals for new services such as VR/AR, achieved with new concepts such as network slicing. This paper is not a complete description of the 5G vision, however the following sections attempt to summarise some of the key components.
Performance & Latency

One of the primary visions for any new network technology is improved performance, and 5G is no different. One of the key areas that is being addressed however, particularly in the RAN, is latency targets, which are designed to be anything from 10-50x lower than in 4G LTE networks.

The targets for performance and latency actually vary somewhat depending on the use-case, of which there are three defined for 5G:

• Enhanced Mobile Broadband (eMBB) - concentrating on high data rates and high mobility up to 500 kmph.
• Massive Machine Type Communications (mMTC) – concentrating on high device density, low data rate, long range and low cost
• Ultra Reliability and Low Latency Communications – concentrating on applications requiring low-data rate but highly available and responsive (low latency) networks.

The following table summarizes the targets for performance, latency and connection density between 4G and 5G networks:

<table>
<thead>
<tr>
<th></th>
<th>4G LTE</th>
<th>5G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Data Rate</td>
<td>25Mb/s</td>
<td>100Mb/s</td>
</tr>
<tr>
<td>Peak Data Rate</td>
<td>150Mb/s</td>
<td>20,000Mb/s</td>
</tr>
<tr>
<td>User-Plane Latency (Air Interface)</td>
<td>50ms</td>
<td>1ms (uRLLC) - 4ms (eMBB)</td>
</tr>
<tr>
<td>Connection Density</td>
<td>2000/km²</td>
<td>1,000,000/km²</td>
</tr>
</tbody>
</table>

The above table shows that although performance does improve, particularly peak performance, it is the network latency which has the most dramatic benefit from moving to 5G.

VNF and Cloud Native Architecture

One of the biggest differences between the 4G and 5G visions is for the deployment of the network components and services. 5G envisages a fully cloud-native architecture, where all functions are provided by Virtual Network Functions (VNFs). However according to the 5GPPP Working Group, traditional NFV infrastructure based on full VMs implementing VNFs may not be adequate to cover all the requirements of 5G networks such as network slicing.

Some have coined the term Cloud-Native Network Functions (CNFs) to describe the additional requirements. CNFs are an evolution of Virtual Network Functions (VNFs), in that they exhibit the following additional properties:

• Microservice rather than monolithic
• Dynamic elasticity and scale
• Deployed and managed using DevOps principles and software
• Containerized – using OS-level virtualization, which may be running on VMs or bare-metal. Examples are Docker/Kubernetes but also systemdspawn containers for example.

CNFs may also benefit from mechanisms such as Continuous Delivery, where the latest updates are deployed automatically rather than in maintenance windows.

Such functions, whether VNFs or CNFs, would be managed by an extension of the ETSI/NFV Management and Orchestration functions (MANO), as part of the management plane.

---

**Edge Computing**

An increasing number of applications, especially in verticals such as retail, manufacturing, transportation and warehousing and health are moving larger part of their functionality to cloud computing. This has many benefits, however many applications require the cloud computing resources to be accessed with extremely low latency. Even with the latency improvement of 5G, if cloud resources are located in the network core or even off-network, those latency improvements are lost.

However if cloud computing resources and applications that run on them are brought closer to the edge of the network, those latency benefits can be retained. This implies edge computing for applications as close as possible to the radio network; even located at the base of the cell tower. Not only does this reduce latency, it also reduces the amount of traffic backhauled across the core network for the mobile operator.

**Network Slicing**

The premise of network slicing is the creation of multiple virtual networks on top of a single physical infrastructure. It leverages the capabilities of SDN and NFV to partition the network into virtual elements, which can then be combined in the most appropriate way to address the use case for a particular segment. By devoting resources in a different way to different slices, the performance, latency, range and other characteristics of the network can vary from slice to slice, and can change dynamically based on the operator's requirements.

Network slicing can be used to serve the requirements of different verticals in an optimised manner, without requiring separate physical network infrastructure.

Network slicing is shown in the following figure:

![Network Slicing Diagram](https://example.com/network-slicing.png)

**IoT**

One of the primary expectations for 5G is the explosion of IoT devices expected to be connected via mobile technology, with verticals such as connected cars, healthcare, industrial facilities expected to be particularly important. The mMTC and uRLLC use-cases will help to drive this growth. It is estimated that by 2025, over 5 billion IoT devices* will be connected over a cellular network.

These new verticals that 5G will enable are quite different from the IoT of today such as lightbulbs and Smart TVs; many of the devices will be performing mission-critical activities, such as controlling or sensing industrial or healthcare equipment or vehicles.

---

A major concern for many operators is the security of such devices; IoT devices have traditionally been very vulnerable to compromise, as well as infection by malware. Compromised or malware infected devices can cause enormous damage, including DDoS attacks, leakage or exfiltration of confidential or sensitive data, or being used to cause physical damage to industrial machinery, patients etc.

Thus many operators consider that detecting and alerting when IoT devices are potentially infected with malware will be a crucial 5G network service, particularly when the fact that IoT devices typically cannot have AV software installed is considered. The operators of such devices would also have a requirement to be alerted when devices are compromised with malware.

Security & Privacy

There are various security realms defined for 5G. For the purposes of this document, we discuss only the user security realm.

5G provides encryption of user-plane communications over the RAN, providing confidentiality and integrity of data transmitted over the air. However this is typically what is known as hop-by-hop encryption, and does not provide for end-to-end (or service-level) encryption between the application representing the user and the service to which they are communicating. Such end-to-end security is typically provided by the applications and services themselves.

Due to the virtualized nature of the 5G core network, the same physical hardware can be used to transport communications across multiple separate network slices; this brings up potential issues of privacy and passive interception if the communication is not secured end-to-end.

3. Why DNS is Crucial to Achieving the 5G Vision

How DNS Fits into a 5G Architecture

Although DNS is often seen as a control plane element, as it is essential to almost all network communication, in the 5G architecture DNS forms part of the user plane. Even though this distinction is somewhat artificial, because provisioning of the DNS service on the mobile access device is performed as part of the initial network attachment, and without it almost all network access would cease to operate. Thus DNS can also be seen as a control plane feature that happens to function in the user plane. To complicate matters further, DNS is also an integral function for the management and control planes, although this would typically be provided by a separate DNS service that is not user-plane facing.

Communication with DNS servers to resolve domain names into IP addresses is typically one of the first actions that is taken in order for two internet endpoints to communicate. There are almost no network connections that take place without an initial DNS lookup, thus DNS is absolutely critical to the operation of IP-based network services.

DNS is Crucial for Achieving 5G Latency Goals

The importance of DNS to overall network performance can thus not be overstated; if a DNS lookup is slow to return then the network will “feel” slow to end users; even M2M communications that do not involve end-users may require time critical connectivity. This fact is often overlooked, which is difficult to understand given the significant investment in mobile network equipment (not just 5G), and the comparatively low cost of a well-run, highly performant DNS service.

Prior to 5G, the impact of DNS latency was less critical; if the RAN only provides a latency guarantee of 50ms, then the additional impact of DNS latency is somewhat hidden within the overall network latency. However with 5G, that situation changes enormously. The user plane latency guarantee for the air interface is between 1ms (uRLLC) and 4ms (eMBB). The latency to services located deep within the core network is likely to be higher, (the distance involved mean that the speed of light comes into play), which is why edge computing is expected to play such a big part in 5G networks.
Let us consider the impact of DNS latency on the eMBB use-case firstly:

- Air Interface latency is around 4ms
- Assume a highly efficient DNS service that delivers results out of a cache for a total internal latency of 1ms
- DNS located in the core network
- Assume an average distance from the cell tower to the DNS service of 500km
- Assume an equipment latency of 2ms (includes all hops)
- Network latency between cell tower and DNS service is approx. 11ms (based primarily on distance, speed of light etc.)
- Additional latency introduced by DNS is 12ms

The additional latency of 12ms is $3 \times$ the latency of the air interface, which means that the minimum realistic latency experienced by a user, application, IoT device or M2M device when setting up a new connection would be 16ms.

The above example assumes a small-medium sized country (e.g. UK sized) with 2-3 regional data centres hosting the DNS service. For a large country the distances involved would be much greater and thus the latencies higher unless additional data centres are added. Conversely for a small country the latency would reduce somewhat. Additionally the internal DNS latency figure of 1ms is a best-case figure, which many DNS deployments fail to achieve.

If we examine the uRLLC use-case, the impact of DNS performance becomes even more stark. If we make the same assumptions as above, we get:

The additional latency of 12ms is $12 \times$ the latency of the air interface, which means that the minimum realistic latency experienced by a user, application, IoT device or M2M device when setting up a new connection would be 13ms.

The conclusion to draw from the above is that if DNS is not to become a latency bottleneck, the DNS service must be located as close to the edge of the network as practically possible. This implies a very different architecture than the traditional regional data centre model that makes up the majority of DNS deployments. Instead, an edge-computing style approach would provide the best latency and prevent DNS from being the weak point in the 5G performance and latency promises.

The following figure shows the latency difference between DNS servers located near the air interface vs those located in the core network:

**Caching, CDNs and Delivering Fast Local Content**

In the enhanced Mobile Broadband use-case (eMBB), a key component of enhancing the user experience is delivering fast content to users. The most typical use-case for end-users is video streaming services, where video streaming should be expected to start almost instantaneously. An equally important use-case for the network operator is to minimise the back-haul of streaming content over the core network; not only does this reduce the performance and increase latency to the customer, but it also increases costs significantly

The way to solve both of these use-cases is to locate streaming content as close to the end-user as possible, typically near the edge of the network. This is achieved by working with streaming content providers and CDNs to locate their content inside the operator network, in data centres that are located close to end-users.

*https://ieeexplore.ieee.org/document/6881230*
The problem that still has to be solved however, is how to ensure that the user ends up being directed to the most local content server. This is typically achieved with either GeoIP (i.e. the streaming service calculates the most likely location of the customer using databases which map IP addresses to location), or using DNS.

The DNS method is the mechanism used by many CDNs, and consists of the DNS service passing information about the approximate location of the user to the CDN as part of DNS lookups; this can be achieved using several techniques, including providing a portion of the original IP address (client-subnet), or simply providing a tag which indicates the region that the customer is located in. With this information the streaming service or CDN can return DNS addresses of services that are located as close as possible to the end-user. Note that not all DNS software supports such tagging of DNS lookups.

There are several reasons why the DNS-based approach is far superior for mobile networks:

• **NAT** – In most mobile networks, NAT (including CGNAT) is used for all Internet traffic. This means that streaming providers or CDNs relying on the source IP address for location will not function correctly, as the source IP address is that of the NAT/CGNAT server, not the end-user’s IP address (which is normally private).

• **Mobility** – End-users on the move need to be served with content from their nearest current location, which is not inferable from their source IP address (even if the private IP address were known).

Once the DNS service has retrieved the localised answer from the streaming service/CDN, it can cache the result appropriately for the end-user, meaning that future DNS lookups from the same user or other users in the same locality will be delivered from the cache and thus deliver even faster streaming service.

It is clear from the above that a suitably configured DNS service is key to delivering extremely fast, localised streaming (and other) content, and that the DNS software must be optimised to work with CDNs to provide locality information.

The following figure shows a non-optimised DNS service with no CDN support, located after NAT/CGNAT:

The following figure shows an optimised DNS service, with support for passing locality to CDNs, and thus providing local content to end-users that does not need to traverse the core network or peering links:
DNS is crucial for Security and Privacy

The 5G RAN uses encryption to secure the air interface, however once the traffic leaves the cell tower it is not encrypted. As discussed earlier, application-level encryption is the best way to ensure that communications between mobile devices and the services they use are not intercepted, monitored or modified.

DNS is no exception – anyone monitoring the network would be able to see all of the DNS lookups that a given end-user or mobile device was making, which is a huge privacy issue as well as a potential security issue if you consider that a MITM attack could also rewrite DNS answers.

There are two technologies that address the security, privacy and integrity issues inherent in traditional plain-text DNS; DNS encryption and DNSSEC.

DNS Encryption

Recently two new standards for encryption of DNS traffic have emerged:

- DNS over TLS (DoT) – Encryption of DNS over a raw TLS-encrypted channel.
- DNS over HTTPS (DoH) – Encryption of DNS over an HTTPS channel.

Both of the above protocols provide similar levels of privacy and integrity protection for DNS traffic. Both are used to protect the traffic between the DNS client and the DNS resolver; future standard will also provide encryption between DNS resolvers and authoritative servers.

These standards are gaining a lot of traction in mobile device software and web browsers. The latest versions of Android will now attempt to create a DoT session before attempting a plaintext DNS session.

Additionally both Google and Mozilla have announced that they will support DoH in upcoming versions of Chrome and Firefox.

Google specifically have stated that they will attempt to use a DoH server if one can be discovered in the local network, but for the time being they will fallback to plaintext if DoH is not available, whilst also making it easy for users to select a different DNS server that does support DoH if they require. However it is likely that Chrome will eventually start warning users about lack of DoH support in the local resolver, which could lead to a wholesale movement of mobile DNS traffic away from the operator’s DNS service to an OTT player such as Google.

Mozilla have already stated that their ambition is to move all their users to one of a trusted set of DoH-supporting resolvers which support strict privacy requirements. The only way to become part of that list is to run a DoH-compliant DNS service.

In the event that mobile operators do not deploy encrypted DNS Services, causing much of the DNS traffic to be routed off-net, the result would not only be a lack of visibility into DNS, but also a worse user-experience in terms of latency and performance (no on-net CDNs), and greatly increased costs for operators, who would have to backhaul non-local traffic across their core network and their peering links.

DNSSEC

Many 5G services, particularly those involving mission critical IoT devices such as connected cars, healthcare devices etc. have a strong requirement to ensure the integrity of DNS answers provided by the network, to prevent hijacking or cache-poisoning attacks, which can lead to private or highly sensitive data being leaked to criminals.

DNSSEC is a way to secure the integrity of DNS answers such that DNS clients can trust that the answers they receive are correct and have not been modified by a third-party. To do this requires the DNS Service to support DNSSEC validation.
Virtualization, Orchestration & Control Plane Requirements for DNS

As discussed above, NFV is one of the primary architectural components of a 5G network, enabling network slicing, orchestration of VNFs etc. However, what does NFV mean for a service component such as DNS? What features and functionality must it support?

The requirements can be summarised as follows:

• Virtualisation – The software must be capable of being run on Virtual Machines (VMs). However the requirement goes deeper than that; for example some software manufacturers only support “Virtual Appliances”; this is clearly not consistent with an NFV scenario where the operator would want to tightly control the OS type and version for all services, for security, functionality and stability reasons. Additionally, being “cloud-native” means the ability to run in containerized infrastructure, and conforming to dev-ops principles and integration with dev-ops toolsets.

• Orchestration – Orchestration is sometimes thought of in regards to VM orchestration, where VMs are created and destroyed automatically, perhaps even in response to compute requirements (elastic scaling). However orchestration also means installation, (initial and ongoing) configuration, and bootstrapping the software running on the OS, i.e. the VNF needs to support orchestration of all the components that make up the running service. This might involve API hooks, and/or support for orchestration tools like Ansible.

• Control Plane – Orchestration is one part of the wider “Control Plane”; this involves everything from orchestration to topics such as component and End-To-End service monitoring, integrating with other orchestration stacks, GUIs for administrators to visualise and deploy the service, etc.

Supporting all of the above requirements is more than a tick-box or an “NFV-ready” stamp. It means understanding how a service will be deployed automatically across hundreds or even thousands of nodes, how configuration changes or upgrades can be rolled out seamlessly and without downtime, how the ops team will be alerted to service, performance or latency issues etc.

IoT and Malware Filtering

The importance of IoT to mobile networks, and particularly 5G, cannot be overemphasised. However, as discussed earlier, the threat of malware-infected IoT devices causing damage to physical infrastructure, networks and even human life, is very real.

Given the lack of AV solutions for IoT devices, detecting malware-infected IoT devices in the network core is the only way to ensure that such devices do not continue to cause harm. This is achieved by using regularly-updated threat intelligence feeds which contain information on the IP addresses and hostnames used to host malware Command and Control (C2) servers. By detecting the devices which attempt to connect to known C2 servers, the devices which are potentially infected with malware can be detected. Their network access can then be blocked, and the owner/operator of the device can then be alerted.

There are two options to reliably detect malware in the network:

• DPI – DPI can be used to detect malware by examining traffic for connections to known C2 servers. However this requires all traffic to be routed through DPI servers, which can be extremely expensive, as well as being a potential network bottleneck.

• DNS – DNS servers can be used to perform filtering of DNS queries, looking for known C2 servers in the DNS request or response. This can be an extremely efficient and cost effective approach, as only DNS traffic needs to be filtered. Modern DNS software can perform the filtering as a built-in function. Some software even supports alerting to end-users via mechanisms such as SMS, Email and mobile Push Notifications.
The following figure shows DNS filtering being used to detect and alert when an IoT device is connecting to known malware C2 servers. In this scenario the DNS server can act as a Policy Enforcement Point (PEP) and also block the connection to the C2 server(s), thus disrupting the operation of the malware:

### Recommended DNS Architectures for 5G

It is clear from the above that the following requirements exist for DNS in a 5G environment:

- **Low Latency** – Mandating DNS services located as close as possible to the air interface; this implies a much higher number of smaller DNS servers than is traditional in legacy network.

- **High Localised Cache-Hit Ratio and CDN Support** – DNS servers must pass client-subnet information to CDNs. This requires caches to be localised as close as possible to the end-user to avoid cache fragmentation, which ties-in with the latency requirement.

- **Encryption** – Support for DNS over TLS and DNS over HTTPS, with a roadmap for additional features such as qname minimisation and encryption between recursive and authoritative servers.

- **Virtualization, Orchestration & Control Plane Functionality** – The DNS service must be a true VNF at a minimum, (ideally CNF), and have built in deployment orchestration and control plane functionality that will interwork with whatever orchestration capabilities chosen by the operator as part of their 5G deployment.

The following figure shows a traditional DNS deployment with a few regional data centres, that may be acceptable for a 3G or 4G network, but does not meet the 5G requirements described in this whitepaper:
The following figure shows a 5G DNS deployment that meets the requirements for DNS in a 5G network:

PowerDNS and 5G

About PowerDNS

PowerDNS was founded in 1999, and has been a leading Open-Source provider of DNS software since 2001. Headquartered in the Netherlands, PowerDNS is part of the OX Group of companies, which are dedicated to making the Internet Open, Safe and Free.

PowerDNS is focused entirely on large-scale DNS service-providers, including mobile and fixed-line broadband operators, hosting providers and cloud service providers.

PowerDNS Recursor and DNSdist

Unlike other DNS resolvers, PowerDNS provides not only an extremely powerful caching resolver, but also a unique DNS proxy and load-balancer called DNSdist. These are normally deployed together to provide an unrivalled feature set for DNS service.

PowerDNS Recursor is a modern, high-performance DNS caching resolver, offering key features such as:

- Highly-Multithreaded, optimizing usage on modern multi-core hardware
- Extremely low and predictable latency for records delivered from the cache
- DNSSEC validation
- Lua policy engine for ultimate customizability and flexibility
- Malware and Content Filtering Engine (including Parental Controls)
- End-User Alerting and Reporting Framework
- Query Logging and Reporting DB
DNSdist is a uniquely powerful DNS proxy, offering such features as:

- DNS-aware load balancing using a variety of balancing and high availability techniques
- Extremely Rich Lua-based Policy Engine
- DDoS Protection
- DNS Tunneling and Exfiltration detection
- DNS over TLS and DNS over HTTPS support
- DNS Query Packet Caching
- Policy-based Query routing

DNSdist is unique in that it can be placed in front of any DNS resolver (including legacy resolvers) to benefit from the above features.

**PowerDNS Performance and Latency**

As discussed at length in this whitepaper, latency is one of the key metrics for optimal DNS (and thus overall network) performance.

The below graph shows data from a real-world production PowerDNS resolver service handling millions of queries per second in a deployment with 2ms average network latency.

The graph, which uses a logarithmic scale, shows that 90% of all queries are answered in 2ms or less, with 99% answered in less than 5ms. The remaining queries are for cache-misses, and thus require a DNS lookup to an authoritative server.

**PowerDNS Caching and CDN Support**

PowerDNS recursor combined with DNSdist provides some extremely advanced caching capabilities, for example:

- Load Balancing DNS queries to resolver pools based on arbitrary information in the DNS packet, including query name, query type, source IP address etc., in order to optimise cache hits or to shard cached queries by resolver.

- Tiered caching – For example using a small cache (including caching client-subnet responses from CDNs) in Edge DNS servers, and forwarding queries to a pool of servers with a large cache (which are configured not to cache client subnet responses). This provides an optimal balance between fast localised DNS responses and minimising latency for domains that are looked up less frequently.
Support for CDNs is twofold:

1. EDNS client-subnet support – Passing information about the original IP address to the downstream server. This is supported in both recursor and DNSdist. The client-subnet-specific information returned by the CDN can be cached or not, depending on the configuration and requirements. Caching is performed on a per-subnet basis to ensure integrity of answers specific to each subnet.

2. Locality Tagging – In an Edge DNS deployment, where the locality is “known”, then a specific client-subnet can be configured for all requests from a recursor. This allows caching to be performed in the same manner for all requests, which means a much higher cache hit-rate.

**PowerDNS Filtering**

PowerDNS recursor includes extremely powerful malware and content-filtering controls, including:

- System-Wide or Per-User Malware filtering – Including Phishing, Malware and Botnet C&C as separate categories.
- System-Wide or Per-User Content Filtering – Based on an internet categorization feed, this can be used to implement network-based Parental Controls or Enterprise Content-Filtering.
- Malware and Content Categorization feeds can be optionally bundled, but also custom or third-party feeds can easily be integrated.
- Alerting/Notification Support – Via SMS, Email, Push Notifications to end-users, or Webhooks to operators.
- End-User Facing REST-APIs – Including OAUTH support, to build mobile apps to control settings and receive notifications.

**PowerDNS Security and Privacy**

PowerDNS supports the latest standards in encryption and privacy:

- DNS over HTTPs (DoH) – Encryption of DNS queries between devices and the resolver
- DNS over TLS (DoT) – Encryption of DNS queries between devices and the resolver
- DNSSEC – Validation of DNSSEC protected records for enhanced integrity of DNS responses.
The following features are on the Roadmap:

- Qname minimization – Preventing leakage of extraneous query information to authoritative servers.
- DNS over TLS – Encryption of DNS queries between the resolver and the authoritative server.

The DoT and DoH support is part of DNSdist, which means that DNSdist can be used to add encryption support to legacy DNS services, as shown in the following figure:

---

**PowerDNS Orchestration and Control Plane**

PowerDNS is comprised of discrete components, each of which can be managed, deployed and scaled separately, for a microservices-style architecture. Every component is packaged using standard Linux packaging tools such as rpm or Debian, and can run on physical hardware, VMs, or containers (e.g. Docker or systemd-nsspawn), i.e. every component is capable of operating as a full VNF.

Every PowerDNS component includes full Ansible automation as part of the software packages, enabling the following functionality:

- Deployment Automation – Automatic and automated deployment of the software to any number of hosts, VMs, containers etc.
- Elastic Scaling – Provisioning and deleting new services as required to support increases or decreases in demand.
- Configuration Management – Automated push of configuration changes to any/all/selected services.
- PowerDNS has been tested and proven with orchestration systems such as Ericsson Cloud Manager.

Monitoring/Reporting is another crucial component of the control plane, and PowerDNS has extensive support for monitoring and reporting on the service, including:

- Prometheus API endpoints for performance statistics retrieval per server
- Metronome support
- Long-term query logging and searching
- End-To-End Performance Tools – for full service availability and performance/latency measurements

The roadmap for PowerDNS includes additional control plane features such as:

1. Configuration Checking – APIs to check configuration changes for validity before changes are made.
2. Orchestration APIs – More native APIs to allow orchestration via configuration management via tools other than Ansible.
Conclusion

This whitepaper has summarized the vision for 5G network services, and how a new approach to DNS architecture and software is required to meet that vision.

The criticality of DNS to the following areas was addressed:

- Latency and Performance
- Delivering Fast Local Content/Edge Computing
- Security and Privacy
- Virtualization, Containerization and Orchestration
- IoT Security

Finally the capabilities of the PowerDNS software were presented, and their applicability to the 5G requirements for DNS were discussed.