CASE STUDY 1

BILLING AND CHARGING PLATFORM EVOLUTION

ADVANCEDIP
intelligent communications

Advanced IP was involved with a major **UK** based telecommunications company in upgrading a legacy system and ensuring zero downtime, increasing availability, security and data integrity all the while maintaining communications with existing legacy systems and 3rd parties. This system generates circa **£1.2bn** in revenue for our customer and was delivered over a period of **18** months.

THIS WAS DONE BY A TEAM OF 3 PEOPLE AND INVOLVED THE FOLLOWING:

SECURITY AND INTEGRITY

Ensure any user based access to the upgraded system is easily identifiable to the user level using RSA Secure-ID.

All revenue and machine based communication is verified against a list of approved firewall rule sets.

All third party connections were Secure-ID authenticated (user identification) and data travelled either via a private leased line or using a site to site VPN.

Where appropriate, a minimum of TLS1.2 level encryption was implemented for web browser security.

AVAILABILITY AND RESILIENCY

All services to the legacy system and new system are fronted by geo-resilient loadbalancers to facilitate automated failover in case of an issue.

MAJOR CHALLENGES FACED

The legacy system used Oracle 9i as a backend database which had to be fronted by loadbalancers. In short, **TNS** in Oracle 9i works in a rather archaic fashion. It initiates a connection from client to server on tcp/1521 but the server responds by stating that a new **TCP** connection should be established using a random **TCP** port number for a data channel as well as stating its own real IP address to be used for that data channel rather than the **VIP.** On the face of it this may not sound like a challenge but with a service fronted by a firewall and a loadbalancer, things can get tricky. We had to create a very specific iRule that carried out a binary search and replace to place the VIP address within the TNS reply (instead of the real server IP) and configured a **WILDCARD VIP** on the load balancers to address the random TCP port number. To address the security aspect of this WILDCARD VIP, we ensured FW rules were locked down to certain sources and enabled ***SQLNET ALG*** on the rule to ensure the firewalls dynamically open ports for random TCP numbers rather than having a static rule which allows all port numbers.

PROCESS

The system in question was around 15 years old with very little documentation attached to it. This caused a significant issue for us as we had to first embark upon a discovery exercise to find all key stakeholders, customers and suppliers of the system. This took approximately **3 – 4** months and was done by analysing firewall log entries, analysing server application configuration and user accounts to identify all parties. Once everyone was identified; in parallel, a new secure **DMZ** was created within the network with Layer **4-7** application and security services and the new service placed within it in parallel to the legacy system. A Virtual IP was fronted before the legacy system and we requested all customers and 3rd parties to use this Virtual address. Once the new server infrastructure was ready, we simply ‘flipped’ the virtual IP address to start using the new servers. This ensured a smooth migration to a new system while maintaining secure communications with third parties and legacy systems.

CASE STUDY 2

RADIO ACCESS NETWORK MONITORING

ADVANCEDIP
intelligent communications

Back in 2014, our major telecoms customer saw a need for a transformational, organisational change in the way in which it monitors 4G traffic and to ensure they meet customer SLAs in key fields. Their current monitoring solution just couldn't keep pace with the high bandwidth needs and rich data that 4G enables in a network. For this reason, they drastically changed their monitoring solution and sent out an RFP for a new vendor. An American vendor was chosen as a best fit and it was down to Advanced IP to ensure the accompanying network solution is fit for purpose for a MINIMUM of five years from day 0.

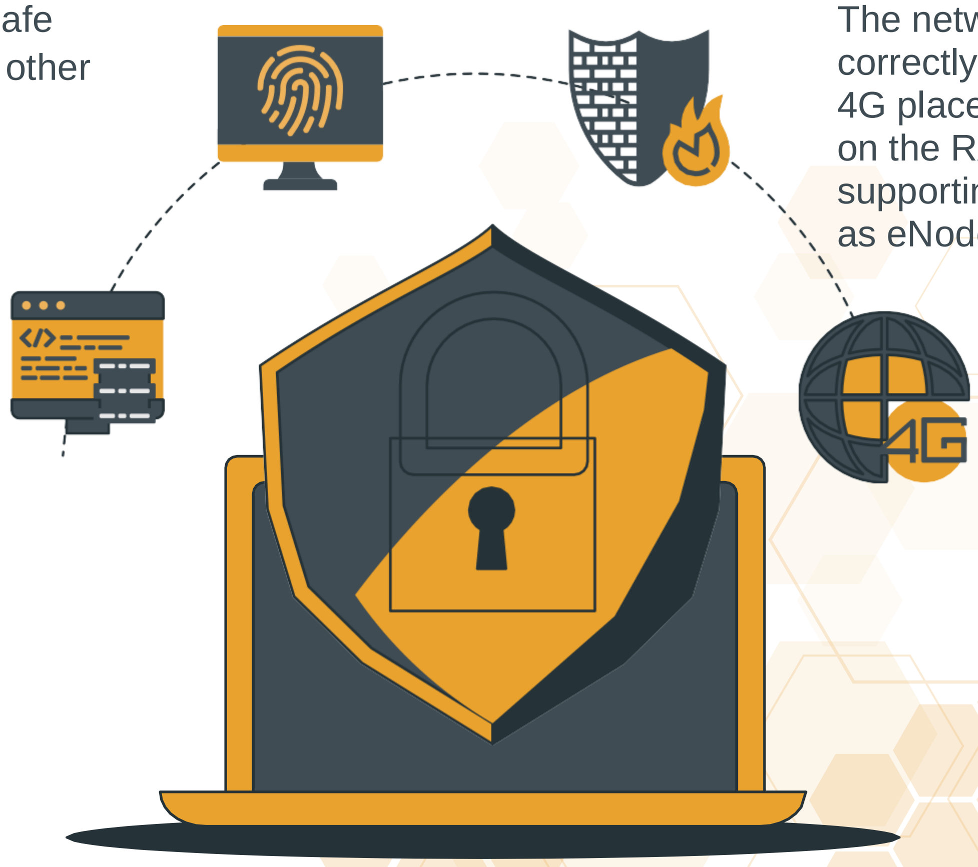
KEY REQUIREMENTS

All user based traffic was to be authenticated using RSA Secure-ID token.

Very stringent firewall rule set for incoming and outgoing traffic.

Traffic must be safe and secure from other telecoms user, application and core traffic.

The network must correctly shape traffic as 4G placed huge demands on the RAN and supporting devices such as eNodeBs, EPCs etc.



INITIAL CHALLENGES

Lack of **10GB** interface support in remote switch sites.

High bandwidth required over the network.

Entirely new infrastructure required to support new monitoring services.

RESOLUTIONS TO ABOVE CHALLENGES

Point 1 and 3 go hand in hand and were resolved by purchasing

- Best in class ASR1001X routing platform at each site for WAN routing
- Nexus 7K for core switching at the main monitoring site
- Nexus 5K and 2K FEX for distribution and access layer
- F5 networks loadbalancers to facilitate high availability
- Check Point Firewalls to provide the best possible security

Point 2 was resolved by marking all traffic before it went out to the MPLS core as CS1/Scavenger class. Once bandwidth upgrades were completed on the core network, our traffic was upgraded to Best Effort with a **25%** bandwidth guarantee.

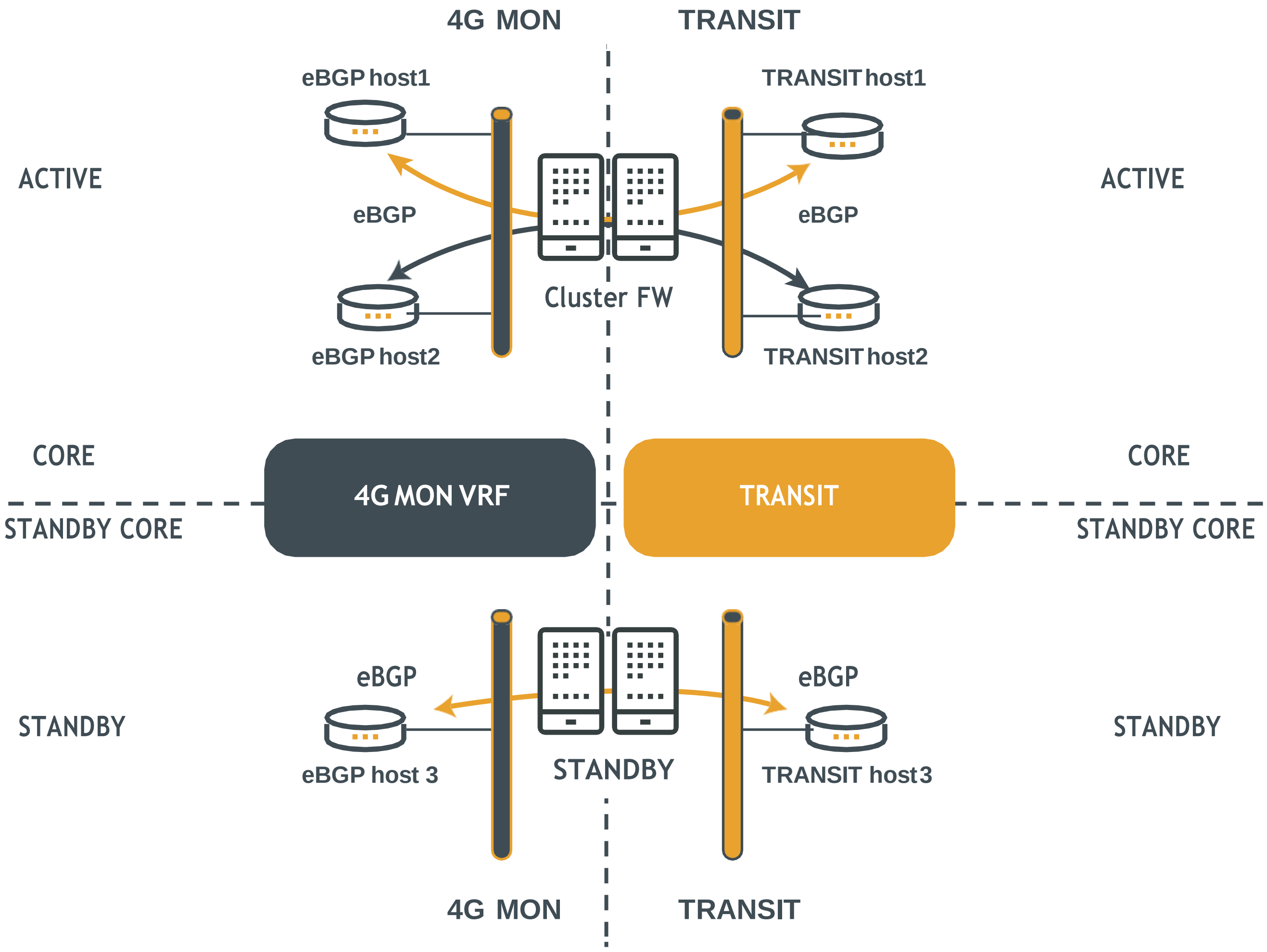
THE SOLUTION

Our solution has to be viewed in a modular fashion and as such, each will be explained as follows:

- **INTER-VRF ROUTING**
- **CORE SITE DESIGN**
- **SATELLITE SITE DESIGN**

INTER-VRF ROUTING

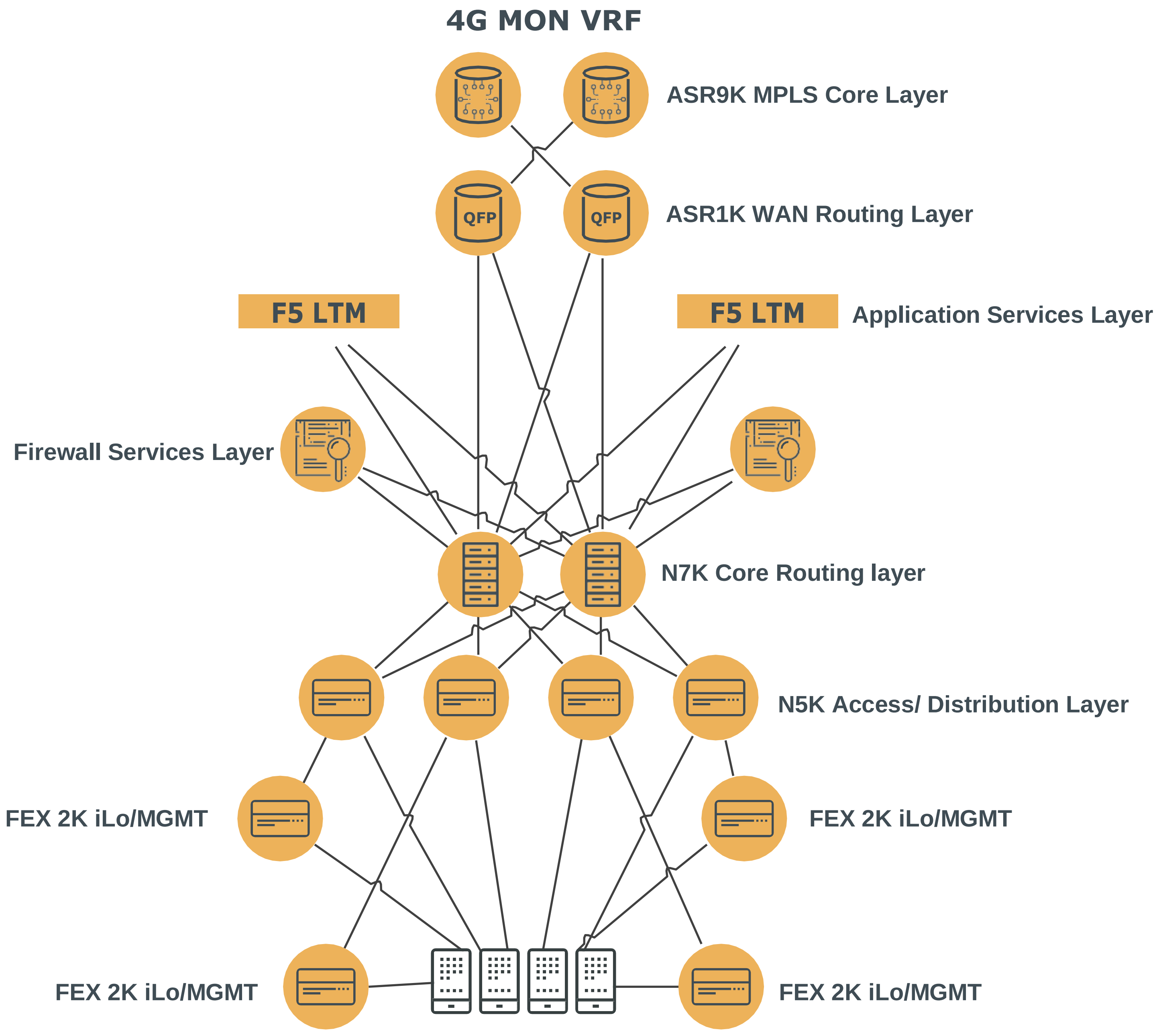
The routing out from the VRF was done using BGP and OSPF. The BGP routers were small routers with a large enough memory to hold about 500 routes. OSPF would hold individual longer prefixes for every site with a static to Null0 for the entire VRF supernet. This supernet would then be advertised into BGP and into the core transit VRF for onward distribution and advertisement. This inter-VRF routing setup was bolted on, along with firewalls at the N7K layer. Inter-VRF routing would only be used for user/admin connectivity into monitoring solution or used for applying patches and updates on all infrastructure, whether server or network. In summary, this was very light weight connection most of the time.



CORE SITE DESIGN

The core site consisted of a pair ASR1001X routers connected into ASR9K MPLS core PE routers at 4 x 10Gbps (2 x 10Gbps per ASR1K) running OSPF. For backend connectivity into the N7K core were more routed ports at 4 x 10Gbps, again, 2 x 10Gbps per ASR1K.

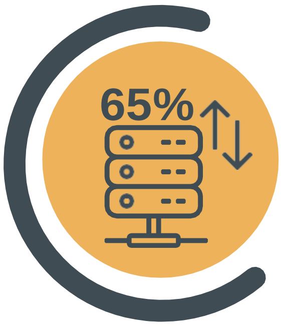
The N7Ks hold the default gateways for all service subnets and route out to other core networks via the ASR1Ks or via the firewalls for inter-VRF routing.



SATELLITE SITE DESIGN

The satellite sites have a very simple design whereby the WAN routing is handled by the ASR1Ks and since there is no need for huge port densities on satellite sites, they have N5Ks and N2Ks to act as access termination points for **10G** and **1G** connectivity.

RESULTS



The project was a huge success for our customer who now enjoy a **65%** reduction in support costs and has fuelled a minimum five year growth forecast for their radio monitoring and support activities.

CASE STUDY 3 LOADBALANCERS AND SSL

ADVANCEDIP
intelligent communications

One of our long standing customers had a very peculiar case floating around within their various support teams… the issue was for every 5 SSL connection attempts made, only 4 would work and go through. Pretty much every single support team had washed their hands of this issue and no-one was willing to take a look. It was finally escalated up to application engineering teams, hardware and software vendors and so forth. Advanced IP were asked to jump in and provide a fresh pair of eyes to look over the various network elements to see if we can root out anything that could contribute to this issue.

THE PROBLEM

Four out of Five SSL enabled connections would work.

THE SETUP

The client was an automated subscriber system that would send customer usage information to a backend system for data processing and financial information gathering. This connection would go through a load balancer and then onto the end platform.

INVESTIGATION

From a network perspective, all routing and anti-spoofing was double checked as well as return path routing to ensure that end to end we had a stable network. Once that was confirmed, we ran packet captures on firewall interfaces to ensure there wasn’t anything out of the ordinary happening on the firewalls such as a silent drop. This was not the case as well so we finally went ahead and proceeded to run captures on the loadbalancers. The WireShark captures showed nothing out the ordinary, low and behold we were in the same position as our customers. We could find no issue so Advanced IP decided to rally the various teams into our Birmingham office and sit face to face and discuss taking this issue further and assign ownership. As a final step attempt, we decided to all troubleshoot together in a room and we finally got to the bottom of the issue!

THE RESOLUTION

In short, the SSL timeout value was too short and stringent for the application and caused instability in the connection. This was picked up because our application colleagues would initiate the connection and THEN gather data before submitting this to the system. The application had a 15min idle timeout applied on the TCP profile so at a TCP level, all would appear normal at the firewall but the accompanying SSL profile was set to 5mins. This would mean that the system would agree an SSL handshake and after around 7-10mins data would be submitted at which point the SSL timeout of 5mins has passed;
1 connection would fail, a new connection would establish and the other four transactions would go through without issue. After increasing the SSL timeout to 15mins to align with TCP we saw that all transactions were now smooth and everything was working as expected!

Advanced IP were asked to advise and present this solution to the design and support teams within our customers organisation and create an SSL design pattern that future designs should adhere to when implementing SSL through a loadbalancer.