SCTE•ISBE CABLE-TEC EXPO® VIRTUAL EXPERIENCE » OCTOBER 12-15 2020

2020 Fall Technical Forum SCTE·ISBE • NCTA • CABLELABS®

Building a Business Service in the Cloud

A Technical Paper prepared for SCTE•ISBE by

Adrian Beaudin
Senior Architect
Akamai
Cambridge, MA
+1 613 670 8451
abeaudin@akamai.com

Bruce Van Nice
Senior Product Marketing Manager
Akamai
Santa Clara, CA
+1 650 381 6074
hvannice@akamai

Table of Contents

1. Introduction

Saturation of traditional markets is forcing ISPs to evaluate new strategies to drive revenues while continuing to deliver a superb user experience and improve returns on capital. Differentiating service offerings is critical to achieving these goals. Commodity connectivity doesn't always motivate subscribers to stay with a service or offer any incentive to pay more for it. Subscriber-facing services that enhance loyalty and increase "stickiness" can help.

Providing managed services and moving up the stack can find a receptive audience amongst business customers. Properly targeted value-add subscriber services can increase ARPU and differentiation and improve retention. But mitigating the costs of this specialty care will drive the use of more efficient, deployment solutions. One of the challenges delivering subscriber services is controlling costs to maintain margins in a world where specialized staff to oversee service deployment and operation are costly and difficult to find. Cloud services deserve consideration to reconcile these conflicting objectives

Cloud services can help contain costs and improve service agility (time to market). They've become mainstream for enterprises, but ISP service enablement in the cloud is a different proposition than typical enterprise IT applications. This paper will offer perspectives from the developer of a cloud service that enables ISPs to deliver security protections for businesses.

2. On the Path to Cloud

Starting in early 2013 a team at Nominum (acquired by Akamai in 2017) started down the path of building a cloud-based security service offering for ISPs. The security service consisted of DNS resolvers and a platform that supported functions such as gathering data, distributing policies, and supporting a portal. As with most development projects there was an evolution with several early, admittedly highly tactical, initial steps.

The first effort was to deploy the software components using the services of a public cloud provider. It was activated at a security conference to provide production DNS resolution (rDNS) services on an open network. Integration of dynamic threat intelligence identified devices with bot infections and there was a portal available for attendees to check their status.

Learnings from this pre-proof of concept led to deployments to support customer pilots of the security service. At the time, getting pilots running at ISPs involved lengthy engagements to get equipment and technician/administrator time in labs. Running these early solutions in the public cloud bypassed this time and materials bottleneck and permitted a faster time to market metric.

Some of the pilots in the public cloud were opened up to run limited production network traffic, so the next logical step was to run more scaled production workloads. The first foray was to support a customer who needed a network protection service immediately but had to contend with a 90 day window to procure and provision hardware in their network to support it. Successful execution of this service in the cloud provided validation and the process of formalizing the service began. All of the early experiments yielded useful insights that fed specifications for development teams ongoing work. Migrating functions to the cloud not only yielded efficiency from the network provider, it also removed the need for the IT team to procure, install and manage the servers and apps attendant to those functions.

3. Service Considerations

The first step for the product and development teams was agreeing on the parameters of the "service". The following sections highlight the primary service considerations.

4. Data Plane versus Control Plane

Security is about examining network traffic to determine whether it is malicious or legitimate.

Approaches that operate in the data plane implement inline packet inspection to evaluate traffic. In the past it was possible to get visibility into most of the traffic on a network, but the predominance of encryption has introduced significant limitations. Inspecting packet traffic at line speeds has always been a costly operation, and inspecting encrypted traffic adds even more costs and operational overhead.

DNS filtering is an alternative that operates in the control plane. Incoming subscriber queries can be matched against dynamic threat intelligence provisioned in resolvers, and policies can be applied to manage unwanted traffic. The team recognized attractive scaling capabilities since there is no need for pervasive filtering of network traffic (and decryption) and it can be layered on infrastructure already deployed and managed. Threat coverage can be expanded by selectively forwarding suspicious traffic (typically domain names that point to both malicious and legitimate web resources) to proxies for further inspection. Experience has shown this is a small percentage of traffic, usually around 2%.

An obvious question given the emergence of DNS encryption standards is whether they make DNS traffic opaque to network operators. The new standards, DNS over TLS (DoT) and DNS over HTTPS (DoH) define encrypted transport for queries between stub resolvers implemented in client devices, and resolvers deployed by network operators (ISPs, MNOs, enterprises, Wi-Fi, etc.). Operators of resolvers using encrypted transports still see queries in the clear and services provided by the resolver function as they would with unencrypted transport.

5. Subscriber Experience

A subscriber's Internet experience is closely tied to latency. In this case DNS resolution is an important part of the solution and the network architecture called for resolvers situated at the network edge as close to subscribers as possible to minimize transit delay. They were dimensioned to align expected subscriber densities with resolver performance (queries per second) to rightsized capacity.

Experience with resolution infrastructure at more than 100 large ISPs worldwide for nearly 20 years showed in today's fixed networks (2020) a subscriber account can generate 10,000 queries per day. Growth has averaged about 20% per year. This provided metrics for sizing subscriber demands on resolvers, and along with a best practice targeting QPS performance at 50% CPU capacity to minimize latency under load, and 5X headroom for growth yielded necessary resolver capacity.

Services that can be customized have more potential to actively engage customers and provide an incentive to remain loyal since they've made an investment in configuring the service to meet their unique requirements. A natural extension of filtering to prevent malicious activity is customer defined filters to block unwanted content for families (parental controls) or businesses (Acceptable Use Policies).

A portal user interface is the subscribers window into their household security posture and another contributor to satisfaction with the service and thus value. Key words for the UI team were "rich, relevant, simple, comprehensible". Menus define web filters and integrated device management capabilities

simplify configuration of user/device specific profiles through device discovery, registration, and pairing functions. Reports display Internet usage and security threats deterred. Scaling, especially with personalization, introduced complexity in the portal infrastructure. Underlying cloud services had to support secure access to millions of unique portal instances, as well as distribution of policy/preferences in near real-time to instantiate subscriber preferences, and collection of data to populate displays for each subscriber.

6. Security and Privacy

The team recognized concerns about security have been an inhibitor for adoption of cloud services and designed layers of defenses to protect the different components of the service. Portals for both administrative functions controlled by the ISP (discussed below), and subscriber-facing functions are protected with each providers Identity and Access Management system. A Web Application Firewall (WAF) protects server and user side APIs (discussed below). Perimeter defenses add another layer of protection.

Privacy is a dominant issue virtually everywhere in the world today and especially in developed countries. Privacy regulations in many parts of the world such as the EU General Data Protection Regulations (GDPR) define frameworks for the permissible collection and management of personal data. Security services and similar services like content filtering are subject to privacy regulations and the development team established several overriding principles followed in the design of the system:

- Only data that is necessary to operate the service, and provide reports covering its operation to subscribers, is processed. Data is not retained longer than necessary for any purpose.
- Data is pseudonymized where possible (e.g. subscriber IPs) in ways that render it impossible/impractical for a 3rd party to compromise subscriber privacy.
- Subscribers can guide processing of their personal data by expressing their preferences in an online portal. They can see the results of filtering (processing) at any time.

Additional data governance considerations motivated the definition of data processing and retention policies. Software built into the components of the service encrypts data in motion. AWS Elastic Block Storage (EBS) encrypts data at rest so the whole partition is encrypted. Further, data in many cases data had to be stored and processed in the country where it was collected (data residency) so in-region cloud services are used. Providers access to all of their data at all times on a real time basis was built into the system. A function was also added that allows providers to zero out their data on demand.

End user requests about their data are currently handled by customer Service Representatives (CSRs) and forwarded to support channels. Data is also available in the system for provider operations teams to integrate with in-house systems to automate this function.

7. Integration into Provider Systems

Cloud services simplify deployment of ISP services by off-loading functions that providers would otherwise have to install and maintain in their networks. Integration with provider systems is still required to:

- Provision Subscriber identifiers, typically within an abstract subscriber ID that's not tied to an identifier that might change (like an IP address or MAC address) and that does not reveal Personally Identifiable Information (PII)

- Make policy changes requested by subscribers through a customer service representative (CSR) or self-service by the subscriber too
- Manage IP address updates in real time through RADIUS messages or DHCP logs. This is necessary due to the decision to allow subscribers to configure their unique content filtering preferences. Their policies have to follow them through address changes.
- Support SSO to allow for integration with subscriber and applications portals for CSR (these portals are described below
- Connect to data collection and management systems. There's considerable built in flexibility to stream protocol and service logs to a central repository to support internal data collection and reporting functions for:
    - Customer support
    - Service adoption
    - Service utilization
- Enable functions required by BSS such as reporting on service adoption

8. Cloud Considerations

In early 2017 the initial cloud release was ready for production networks. At a high level there were 4 major components to the service to support in the cloud: DNS resolution, data transport, policy management, and subscriber and operator interfaces. Teams first explored how to scale the service in the cloud for daily usage patterns and peak events while accommodating growth in demand. Established metrics for the resolution component, discussed above, were extrapolated to build out the other components of the system.

Given the criticality of availability/reliability with respect to user experience a decision was made to deploy a single stack for each provider, versus taking advantage of potential gains from multi-tenancy (although these were never really measured so it's not clear they exist for this kind of service). Isolation has numerous positive implications, one customer can't take down the service, performance is more predictable, there's less data/privacy exposure and data can be repatriated. For resilience the service is deployed in fully redundant stacks in two different public cloud regions. A single region or service will not create an outage.

A Service Level Agreement tracks average availability on a monthly basis from telemetry data available through administrative service interfaces. Service credits are calculated when outages exceed an availability baseline.

The service was also designed so providers could deploy some components in their own networks in a hybrid configuration, or deploy the whole stack.

9. APIs

Early production deployments of the cloud service quickly highlighted features and functions that were necessary and not yet implemented. APIs rose to the top as essential. Provisioning interfaces were needed to integrate with provider operational processes and systems. Two provisioning APIs were created, one an abstraction and one low level. AWS APIs were a source of inspiration and insights for the development teams and used as the model for their efforts. Providers should hold all their cloud service vendors to a similarly high bar!

One of the provisioning APIs powered the subscriber portal so providers can build customized, purpose-built portals and match the user interface look and feel to other portals subscribers have access to. This API was beneficial from a vendor perspective as well since it reduced the burden developing differentiated features requested by individual customers.

The ability to provision subscribers/services was also exposed in this API - objects that represent subscribers and the services they have could be created. Portal authentication integrates with AAA such as RADIUS, LDAP can use SSO. A third API was created for IP to subscriber mapping using dhcp log scraping, radius accounting message ingestion or a REST API.

Several requirements were established for APIs

Proper documentation - Use swagger to generate docs in its standard format. Offer pdf docs with integration examples.

- Versioning. Enable critically important backwards compatibility for integrations.
- Simple API calls. Make heavy use of abstraction to minimize complexity.
- Read (GET) calls. Verify subscribers are properly provisioned in scaled, high throughput installations.

10. Customer Support and Operations

Customer support functions typically take advantage of APIs for subscriber provisioning and IP tracking, supplemented with logging information gathered by the service. A portal was designed to allow CSRs to look up subscribers, and their configuration (IP addresses, policies) as well as when they last generated traffic with the service. A pre-built portal was created, and APIs can be used to create a custom portal.

Another portal was created to provide global reports on the service such as Top infections, Top blocks, and service adoption. This reporting data can also be integrated with BSS or other custom interfaces used by a provider. Access to both portals is controlled with RBAC and can be integrated with an external directory such as LDAP.

11. Summary

Value-add subscriber services are becoming more strategic as ISPs look for new ways to grow revenues. Controlling costs for service enablement is essential, and reducing time to market is increasingly mandated, even as it's harder to find and retain specialized staff, and rigid processes to deploy new functions raise costs and inhibit agility.

Widespread availability of higher broadband speeds to both enterprises and residential customers has greatly facilitated the feasibility of control or data plane, cloud-based services. When typical customer locations have access to 100Mbps of bandwidth or greater, even with conventional latency the performance is high enough to allow remote and cloud compute as part of routine communications.

For these reasons cloud services deserve consideration to reconcile these conflicting objectives. They've become mainstream for enterprises, but ISP service enablement in the cloud is a different proposition than typical enterprise IT applications.

Product and development teams at Akamai created a cloud based security service designed to make it easy for ISPs to target customers with a new offer. Focus and 20 years' experience drove design considerations to:

- Enable the best possible subscriber experience, minimizing latency, providing a high degree of customization, and effective security with “just works” simplicity
- Build scale, availability and resilience into cloud-based components that support the service
- Ensure security and respect for privacy for the service itself, staff who operate it, and subscribers who use it
- Integrate with provider systems to automate provisioning, support monitoring, and collect data for business systems and to meet regulatory requirements