HOLESHOT

NEXT-GENERATION IMAGE EXPLOITATION

ABSTRACT

Your enterprise is being inundated with valuable intelligence, hidden away in an onslaught of thousands of massive images. Machine Analytics and Human analysts alike struggle to get full access to their areas of interest in a timely fashion. HOLESHOT tackles this problem by recognizing usage trends to performantly deliver mission critical pixels to the right users in a significantly faster and more efficient manner; delivering increased performance to the mission while decreasing downrange storage and bandwidth utilization by 80% over today's systems. HOLESHOT also provides an open-source/SIPS-compliant ELT that delivers world-class imagery exploitation at a savings of over $50K-per-seat.

TABLE OF CONTENTS

PROBLEM STATEMENT

A single image in your enterprise can reach a size of over 40GB, and the amount that are taken every day is staggering. Commercial and national assets have a daily imagery take in the millions; the Planet constellation alone is comprised of over 146 smallsats, gathering over 1.4 million images per day which cover over 300 million square kilometers. The current architectural triad used to deliver these images for exploitation (i.e. NCL, CAOS, and IEC), is pushing this imagery to the four corners of the planet via a vast spoke-and-hub topology that replicates the images numerous times along the way. This necessitates massive network pipes and downrange caches to bring the imagery close enough for exploitation to occur in a performant manner. This congested system means delays between time-of-intercept and eyes-on-target; decreasing the mission relevance of the data.

Distribution of this imagery, at this scale, is also fairly unique to the Defense and Intelligence Communities. Housekeeping, security, staging, and automation of these massive images at scale is not something that the rest of the world is burdened with. This has necessitated the maintenance and recapping of custom software and buffers at a significant cost.

When the architectural triad was conceived over 20 years ago, the daily imagery take and the size of the individual images were orders of magnitude smaller, as were the number of sites and customers that the imagery had to be moved to. The architecture simply hasn't scaled to meet the mission needs and capabilities, as well as the fiscal constraints of modern times.

BACKGROUND

In 2016, NGA's eXploit program was charged with providing a COTS-based web application to serve as the next generation ELT for the enterprise. This application was hosted in an AWS cloud environment and served out finished tiles to the user's browser. Unfortunately, it was soon learned that this approach had some significant architectural shortfalls inherent to its design:

- The COTS vendors of web-based ELTs, in an effort to get to market quickly and capitalize upon their existing desktop baselines, had basically ported their desktop variants into a web-server. This meant taking code that was designed for a single user and attempting to scale it to thousands.
- The basic design of the infrastructure for these applications still required full NITF imagery files to be located close to their cloud-based tile servers. This necessitated a managed filesystem-based buffer to exist in a relatively close proximity to the cloud applications. Since eXploit was an ASP, it relied on a yet to be created CSP (i.e. BCN) to be stood up in the same VPC as eXploit to have reasonable performance expectations. Without the CSP, eXploit had to implement a managed buffer of their own in short order. (This common scenario of needing a managed buffer is retold everyplace that imagery is exploited across the enterprise, whether it's an analyst on their Remoteview or a computer vision algorithm trying to do automatic detection of aircraft)
- Tiles were rendered server-side and delivered to the client browser. This meant that every time that an image needed to be adjusted (e.g. Brightness, Gamma, Contrast), all of the finished tiles needed to be recalculated on the server; the client's screen went black for a few seconds, and then slowly filled in. This context-shift caused regular disruptions to the analyst's concentration.
- The finished tiles were unsuitable for consumption by most computer-vision algorithms as critical information is lost as bits-per-pixel are decreased and histograms are shifted into the visible spectrum.

The eXploit architecture team realized that an effective solution would have to deliver full resolution raw image tiles to the clients to get the responsiveness that the analysts had become accustomed to with their desktop systems. This would allow the clients to perform the necessary processing, locally on their own GPUs, drastically decreasing the time from user input to rendered adjustment. The only problem was that the concept of breaking out imagery into raw image tiles and reassembling them client-side in a performant fashion was unproven; there were simply no products in the marketplace to meet this need. There also wasn't an appetite to build a custom solution off of such unproven ideas. The ELT portion of eXploit was discontinued in favor of a new GOSS alternative that had been developed. iSpy provided the same functionality as the COTS products, but at a lower price-point and with higher reliability. Unfortunately, iSpy still suffers from most of the architectural woes that plagued its COTS predecessors.

TARGET SOLUTION

Following the demise of eXploit Core, Leidos invested in proving out the raw-image tile delivery system. We labeled it HOLESHOT and gave it a few core objectives:

- It had to be born in the Cloud, imbuing all the elasticity, scalability, and performance inherent to that environment.
- It had to serve the image exploitation needs of the thousands of concurrent users.
- It had to prove that a client-side ELT, driven by a raw image tiling system, could perform as fast, if not faster than today's best desktop ELTs. Roam, zoom, DRA, and image adjustments would need to provide the same desktop performance as the user's current environment.
- It had to be open-source if it was to succeed. Opaque stovepipes have stymied innovation for too long, limiting the ability for the infrastructure to keep up with the needs.

Three years later, we've not only met our objectives, but have also opened up the door to new architectural and mission capabilities for the enterprise.

WHAT IS HOLESHOT?

HOLESHOT is premised on the postulate that analysts don't look at all of the images that are staged to them. Furthermore, analysts rarely look at more than a small percentage of any given image (e.g. - an Airfield, port, or facility).

Instead of moving entire full-framed images around the enterprise, HOLESHOT moves only the portions of the image that an analyst is likely to look at. This results in an order of magnitude decrease in data transfer and storage downrange.

HOLESHOT attains insight into an analyst's needs by running analytics to generate multi-resolution heatmaps that show their areas of interest. As new imagery arrives, these heatmaps are used to route the appropriate tiles to the appropriate storage tier. Users retain full accessibility to the entire image, but historically accessed areas will have a much faster access rate. This affords the analysts the performance of their traditional desktop systems while drastically decreasing overhead on the enterprise.

On the Cloud-side, imagery is resident in S3 and cached in multiple tiers, including Elasticache and DynamoDB. Downrange, tiles can be pushed via CDNs (e.g. Cloudfront), commodity caches (e.g. Akamai), group shared drives, or local filesystems.

HOW IS HOLESHOT DIFFERENT THAN HISTORIC TILESERVERS?

Traditional tileservers, such as RVCloud, ISpy, and WebGLT, read full-framed raw imagery from a local filesystem and feed out finished tiles at display time.

Roaming performance on the browser is blocky, as areas to render are blacked-out prior to the tile being retrieved from the server; similar behavior is experienced whenever adjustments to imagery are made (e.g. TTC, Gamma, Brightness, etc). These fetching interruptions and context shifts distract the analyst from the exploitation process.

The HOLESHOT architecture takes a drastically different approach. By retaining the full bits-per-pixel and band spectrum of the original images all the way down to the client, HOLESHOT allows clients full access to the data. It optimizes this access through multiple tiers of storage spread across the enterprise whose selection is driven by advanced usage based analytics. Pixels that the user is interested in are moved to the appropriate storage tier to ensure optimal performance. By serving out raw image tiles, the imagery can manipulated by the local GPU, ensuring optimal roam and image adjustment performance, By retaining full pixel resolution, HOLESHOT can meet the needs of ELT's, mashups, and computer vision algorithms; rather than the singular visualization capabilities of traditional tileservers.

HOLESHOT was born in the Cloud and was designed to serve your imagery at scale. Tests have shown it capable of running at over 1500 Tiles-Per-Second, for months at a time, without downtime. The architecture is also capable of pushing out SIPS-Compliant display-ready image tiles at scale, service-enabling web-based mashups, WMS/WMTS/MRF compliant clients, and analytics across the enterprise. These can be served in image space or warped to a projection of the user's choosing.

While clients that are fully integrated with HOLESHOT will achieve the greatest performance levels, we've also created an adapter that allows legacy applications to take advantage of the novel architecture. By creating a virtual filesystem, we've made it possible for legacy applications to believe that they're reading local NITF files, while in reality they're actually reading from the HOLESHOT tile system.

VALUE PROPOSITION

HOLESHOT enables imagery for the enterprise. Raw or SIPS processed imagery is made available in the Cloud and downrange to ELTs, web applications, and analytics. The applications get exactly what they want without the need for custom local buffers, large memory footprints, and large network pipes. Advanced applications get the entirety of the data that they need to do their job. Simpler applications get the convenience of accessing finished imagery without the need for advanced image processing algorithms.

HOLESHOT allows for Centralized/Cloud-based management of all image holdings, increasing security and reliability, while decreasing time and personnel needed for release deployments. It does this while decreasing image network traffic and downrange storage requirements by over 80%.

This architecture is squarely aimed at enabling the retirement of some long-standing systems within your environment; CAOS image orchestration and custom IEC managed Buffers will become unnecessary when the HOLESHOT architecture is fully integrated into the enterprise. Through the elimination of full-framed image staging and its associated timeline; users across the enterprise will be able to access imagery in seconds rather than hours. By breaking down the file paradigm, users will also be able to

roam temporally as easily as they do spatially (i.e. moving through a stack of the enterprise's image holdings without losing geospatial context is as easy as roaming left or right on an image.)

PROPOSED PATH FORWARD

HOLESHOT is a performance enabler that for a relatively small investment could save millions across the course of the next few years. It has been tested out in the commercial cloud feeding demonstrations and proof-of-concepts; most notably our CAF (Consolidated Analytic Framework) demonstration that was met with high-praise at GEOINT 2020. In 2021, we have plans to stand up HOLESHOT for our in-house production environments where it will feed imagery to our cell of in-house analysts, ingesting and serving out thousands of images a day.

The eventual sweet spot in your environment would be for HOLESHOT to be stood up within the ODS enclave for serving out imagery to the greater NGA enterprise. When this happens, it will be clear that numerous legacy systems can be retired.

As an interim step, we recommend standing up HOLESHOT as a study within the SAS program. This study would deploy HOLESHOT into C2S, affording analysts a full evaluation of its performance and a true assessment of its significant potential. We believe that this can be done on a relatively small budget, with the biggest cost going to modifications to the security plans and system accreditation. A formalized estimate of the cost can be provided upon request.

HOLESHOT is an empowering architectural concept that requires government support to get it into government spaces so that it can achieve critical mass. It is the only cloud-born architecture that we're aware of that is attempting to tackle the significant data challenges facing NGA and is completely open-source.