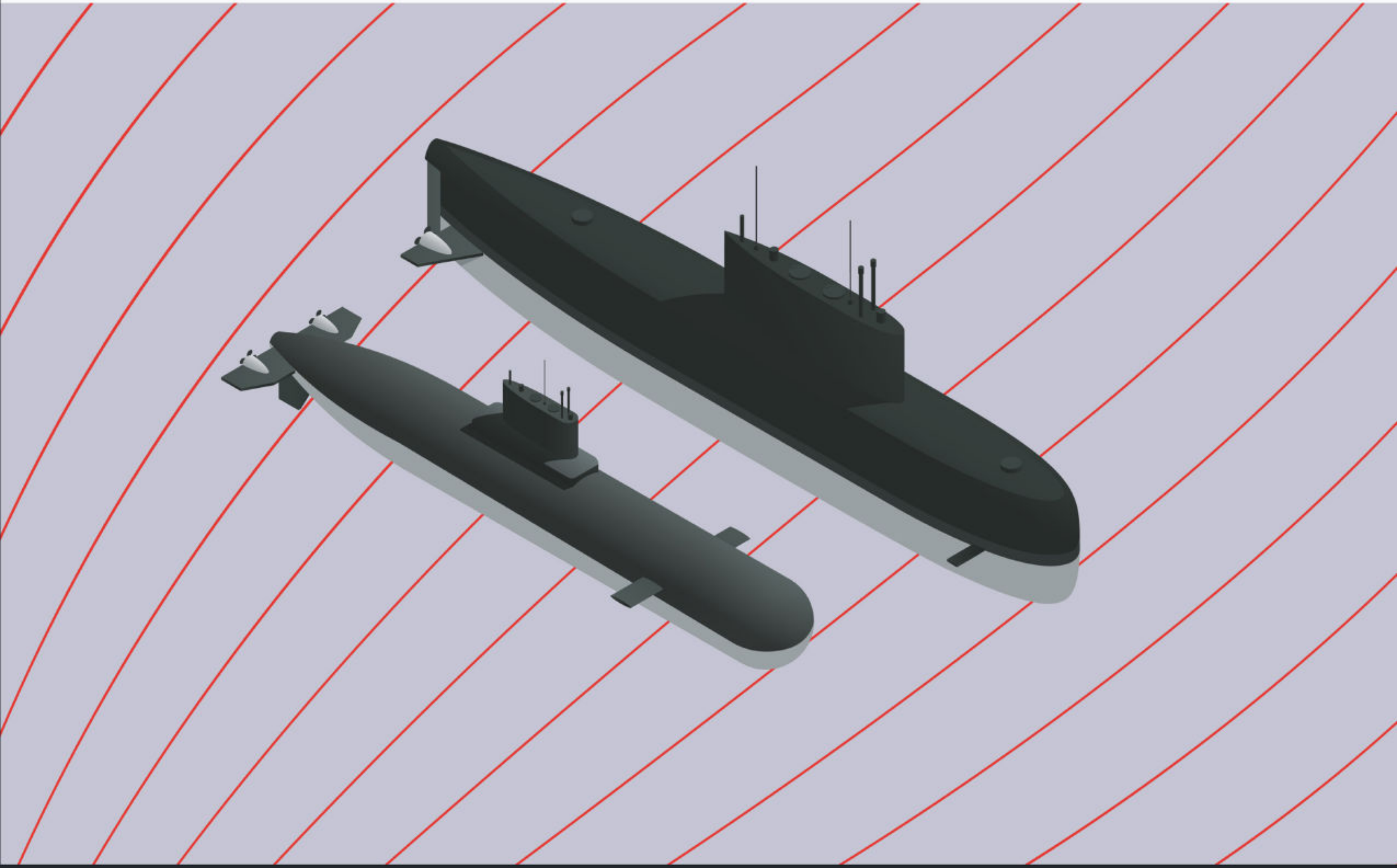
INQUEST



InQuest® RetroHunting

Retrospective Analysis Fueled by Deep File Inspection® (DFI)

InQuest RetroHunting

Retrospective Analysis Fueled by Deep File Inspection (DFI)

Table of Contents

InQuest RetroHunting

Retrospective Analysis Fueled by Deep File Inspection (DFI)

Executive Summary

The purpose of this white paper is to detail the functionality and benefits of the InQuest platform, specifically it's Deep File Inspection (DFI) and RetroHunting capabilities. Fundamentally, DFI empowers defenders with a new dimension of data while RetroHunting exposes a new dimension of time. This novel capability is made possible through a proprietary, carrier-class, network traffic processor coupled with a custom file storage algorithm focused on atomic level deduplication. Regardless of whether or not there is any perceived threat or data-loss event at the time of ingestion, all files and session metadata are inspected and stored for future reinspection. Retrospective analysis is initiated both automatically, on regularly scheduled intervals, and manually by users of the platform. The default platform settings result in three independent analyses of every captured file over as many weeks. In other words, an alert may be generated for data ingested three weeks ago, based on the latest threat intelligence from today.

In the first section, we provide a high-level overview of the InQuest mission, team, and platform. Subsequently, we highlight the platform's data acquisition and storage capabilities. Next, we examine specifics regarding real-time analysis and threat intelligence. Finally, we dive into RetroHunting. Visually, both the InQuest platform and this paper are organized as follows:

VISIBILITY
ANALYSIS
DETECTION

Figure 1: High-Level Data Flow

RetroHunting represents the pinnacle of our offering. As such, it is predicated on how we capture and analyze data. An understanding of how data capture and analysis is performed on the InQuest platform is imperative to the understanding of the capability and value of RetroHunting.

InQuest Company and Solution Overview

The InQuest platform provides high-throughput Deep File Inspection (DFI) for threat detection, data-leakage detection, and threat hunting. We ingest, dissect, and catalog files for real-time and retrospective analysis, leveraging the power of hindsight to apply today's threat intelligence to yesterday's data. Built by SOC analysts for SOC analysts, we empower defenders to save their organizations most precious and limited commodity, human cognition, by democratizing advanced malware analysis skills to reduce analyst fatigue and frustration and increase return on investment with regards to personnel.

We aim to automate and scale the expert knowledge of a typical SOC analyst. Available on-premise or as a service, the InQuest platform leverages a variety of sources in our automated decision-making engine. This includes bi-directional orchestration with multi-scanning and sandbox platforms, unique threat intelligence sources, and a seasoned signature development team augmented by machine learning.

Founded in 2013, the InQuest leadership and engineering teams are comprised of passionate security researchers hailing from both the public and private sectors. Our mission is to deliver our decades of lessons-learned to protect users and organizations everywhere. We strive to maintain our hands-on familiarity with a wide range of prevention, analysis, and monitoring solutions, continuously exploring, examining, and validating security vendors. We continue our community involvement through contributions by way of talks, publications, open-source software, and threat research collaboration. Two unmatched advantages fuel our team's differentiators:

1. We've worked with thousands of real-world exploits from vulnerability discovery and exploitation specialists from all around the world.
2. We've vetted nearly every major security vendor under the Sun, having hands-on experience with the best of breed Commercial Off The Shelf (COTS) and Open Source Software (OSS) solutions across the spectrum.

Regardless of how the InQuest solution is deployed, our goals are to:

1. Reduce analyst frustration and fatigue by acting as a force multiplier to support the needs and scale of businesses ranging from small offices to global enterprises.
2. Expose deeply embedded malicious logic through novel methods, automating typically human-intensive tasks to democratize expert level skill sets to a wider audience.
3. Utilize any and all analyst and threat intelligence resources available in customer environments to automate the identification and validation of threats and data theft.

For further details, or to get in touch, visit us at www.inquest.net.

Data Acquisition

Data Capture

The InQuest platform supports a variety of deployment options and ingestion methods. Available on-premises, virtualized / in-cloud, or as a service (SaaS); they include:

- SPAN / TAP / VPC
- CIFS / NFS / SMB / SSH
- ICAP
- API / Manual Upload

While the primary ingestion artifact is file data, the platform will associate any available session-level metadata (such as mail and web headers) with captured files. The complete pool of captured artifacts is stored, processed, and analyzed. Artifacts include domains, files, hashes, headers, IPs, SSL certificates and URLs.

It is important to note that all files of interest are retained, regardless of whether or not there is any perceived threat or data-loss event at the time of ingestion. Additionally, a number of extensive retention signatures exist for capturing file-less malware, malware pivots, and other suspicious objects in transit. Finally, the entire subclass of extensible header signatures (mail and web) are executed on every captured session and any alerts from this layer also result in retention. See Figure 2 for a visual representation of this concept:

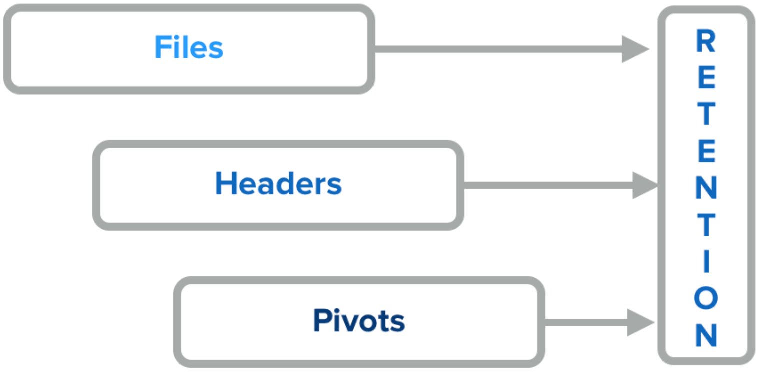


Figure 2: Retention Strategies

Default retained file types include archives, executable formats, and common malware carriers such as Adobe and Microsoft Office documents, Java and Flash applets. While not exhaustive, the following sampling of well-known and commonly recognizable retained file types cover the vast majority of both malicious and benign formats.

Carrier Extensions	Archive Formats	MIME Types
● CHM ● DOC* ● EMF ● EXE ● HTA ● HTML ● JAR ● JS ● LNK ● PDF ● PPT* ● PS1 ● SWF ● WMF ● XLS*	● 7Z ● AR ● ARC ● ARJ ● BZIP2 ● CAB ● COMPRESS ● CPIO ● DEB ● FLAC ● GZIP ● ISO ● LZMA ● RAR ● RPM ● TAR ● XZ ● ZIP	● application/cdf* ● application/java* ● application/msword* ● application/pdf* ● application/vnd* ● application/x-java* ● application/x-shockwave-flash* ● text/rtf*

Table 1: Retention Sampling
*(Note that * indicates a wildcard)*

Data Storage

Session metadata and extracted artifacts are stored in a relational database whereas file artifacts are stored on the filesystem and organized via a proprietary deduplication algorithm we call POUNDfs. As mentioned in the summary, this storage algorithm implements atomic level deduplication to maximize data retention. What this means in practice is that deduplication occurs not only by cryptographic hash for ingested files, but as well as on content derived during the DFI process. If two different files produce similar DFI outputs, storage is shared between them. In deployments with multiple InQuest components, deduplication can be coordinated between them in a mesh pattern, thereby preventing the necessity for two components to store the same file data. For example, if in an on-premises deployment, two separate collectors ingest the same file, only the first collector to "see" that file will be responsible for data storage. The second collector can reference the data from the first collector, directly. As the DFI process typically results in 300% (3x) more data than the amount originally ingested, such deduplication is a requisite facet of the overall system for ensuring optimal retention to maximize the RetroHunt lookback window.

The file retention window is affected by the user-defined policy, mixture of ingested data, and storage capacity of the underlying InQuest component. InQuest collectors are provisioned to support at least 30+ days of file retention and 365+ days of metadata retention at their peak rated speeds. There is no design limitation to data retention and therefore retention rates can be increased by selecting over-provisioned components. For example, placing a 20Gbit collector on a 10Gbit pipe.

Data Flow

Regardless of deployment, the logical data flow within the InQuest platform remains the same: extract artifacts, process files through the DFI engine, provide artifacts to optional third-party / in-cloud services, produce a single all-encompassing threat or data-loss score (ranging from 0 - 10) accompanied by a "receipt" that demonstrates the contributing factors at a glance. Figure 3 depicts a typical on-premises deployment scenario, featuring a 10Gbit InQuest Collector, a 20Gbit InQuest Collector, and an InQuest Manager. The physical appliances are denoted with a red magnifying lens icon, all other glyphs represent logical data flow. The Collectors are headless in the sense that all tuning and querying are accomplished from the Manager. Each Collector is responsible for its own file storage while the Manager is responsible for the storage of metadata across all managed Collectors.

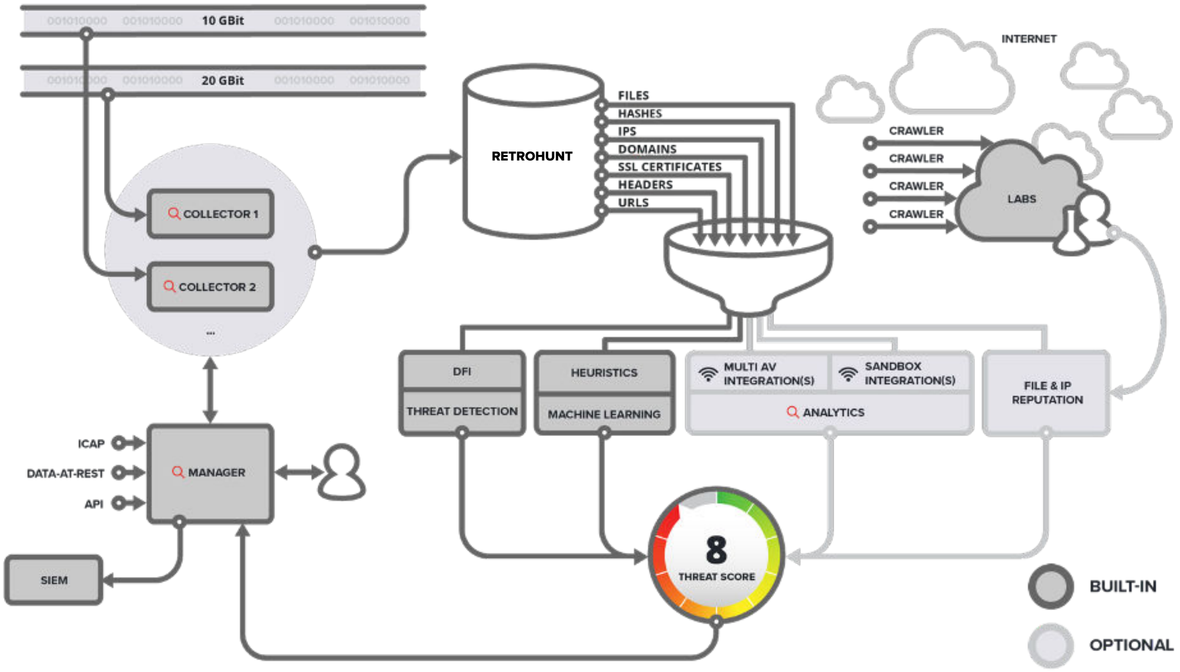


Figure 3: On-Premise Data Flow Diagram

While the InQuest Collectors are dedicated to native packet capture (SPAN/TAP), the InQuest Manager is capable of data ingestion via ICAP, data-at-rest, API, and manual upload. In the next section, we dive more deeply into the Deep File Inspection and real-time analytics data flow path depicted in Figure 3.

Real-Time Analysis

The advent of ubiquitous security controls in both compiler and operating system hardening technologies have altered the threat landscape. Previously, no-interaction, singular vulnerabilities were discovered and exploited with regularity. The modern-day defensive fabric requires attackers to chain together multiple, sometimes over a dozen, exploit pivots to successfully gain control of a targeted system. The increased use of sandbox or detonation solutions has driven attackers to avoid detection through required input or interactivity from the user. In some cases, a number of manual steps must be performed by the user before the underlying payload is activated. Otherwise, it remains dormant and therefore undetectable through behavioral analysis.

It's no secret that client-side attacks are a common source of compromise for many organizations[1]. Web browser and email borne malware campaigns target users by way of phishing, social engineering, and exploitation. Office suites from vendors such as Adobe and Microsoft are ubiquitous and provide a rich and ever-changing attack surface. Poor user awareness and clever social engineering tactics frequently result in users consenting to the execution of malicious embedded logic such as macros, JavaScript, ActionScript, and Java applets. Analysis of these common malware carriers is time consuming, tedious, and requires expert skills. We invented Deep File Inspection (US Patent #US20150281260[2]) as a solution to democratizing and scaling this previously human-intensive process.

Deep File Inspection (DFI)

DFI is a core tenet of our solution. DFI is a static-analysis engine that peers deep beyond layer 7 of the OSI model, essentially automating the mundane tasks of your typical SOC analyst or security researcher. Regardless of the novelty of nesting employed by an attacker, DFI will rapidly dissect common carriers to expose embedded logic (macros, scripts, applets), semantic context (e.g. cells of the spreadsheet, words in a presentation), and metadata (e.g. author, edit time, page count). Images discovered to be embedded are processed through a machine vision layer (OCR, perception hashing), adding to the semantic context extracted from the original file. Common evasive characteristics and encoding mechanisms are automatically discovered and deciphered. The DFI process typically results in three times (300%) the amount of analyzable content. For example, 6MB of data may be derived from a 2MB file, resulting in 8MB of total inspectable content.

A general frustration voiced by SOC analysts and information security researchers is the limited availability of context for detection analytics. In the case of IPS, resources are limited to just microseconds of time and kilobytes of analyzable data. IDS systems can typically delve deeper, taking additional milliseconds of time exposing further data. The next step up, with regards to

[1] https://enterprise.verizon.com/resources/executivebriefs/2019-dbir-executive-brief.pdf
[2] https://patents.google.com/patent/US20150281260

the time-vs-analysis trade-off, is behavioral monitoring or sandboxed execution. This class of solutions detonate samples in a virtualized environment and annotate the behavior of the system for threat detection... this process is both compute and time-intensive, taking, by design, minutes to analyze each sample. The InQuest platform addresses the time-vs-analysis gap with Deep File Inspection (DFI), which typically completes its static analysis in 2-4 seconds and provides megabytes of additional analyzable content through a variety of sources and methods.

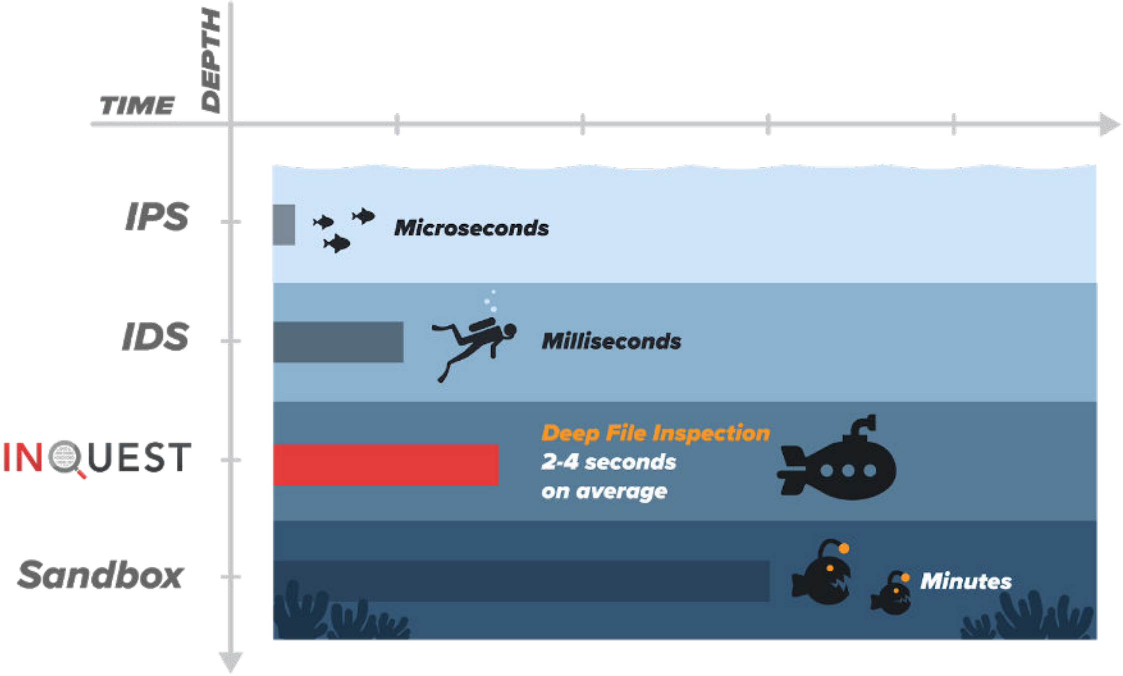


Figure 4: Time vs Depth of Analysis

Figure 4, above, depicts the relationship between time and analysis depth across a variety of solutions. InQuest DFI fills an obvious gap by trading seconds of time for a wealth of additional data-depth and data-normalization, easing the burden of signature development for InQuest Labs analysts, customer SOC analysts, as well as for heuristical signature-less based detection methods. Table 2 shows the typical analysis time and data expansion rates for the most common non-executable MIME types analyzed by the InQuest platform. These statistics were compiled from a corpus of ~1.2M real-world files sourced from the VirusTotal Intelligence feed and harvested since January 1st of 2018. The methodology for collecting these files was detailed by InQuest CTO Pedram Amini[3] at Blackhat USA 2019 in a talk titled "Worm Charming: Harvesting Malware Lures for Fun and Profit[4]", the YARA rules used for collection are available on Github[5].

[3] https://www.blackhat.com/us-19/speakers/Pedram-Amini.html
[4] https://www.blackhat.com/us-19/briefings/schedule/index.html
[5] https://github.com/InQuest/yara-rules/tree/master/labs.inquest.net

MIME Type	File Count	Analysis Time	Data Expansion
application/cdf*	123,340	1.56 seconds	110.89%
application/java*	98,974	2.55 seconds	218.65%
application/msword*	134,683	1.14 seconds	89.64%
application/pdf*	448,507	2.41 seconds	408.30%
application/vnd*	307,399	3.54 seconds	381.58%
application/x-java*	38,162	1.80 seconds	316.31%
application/x-shockwave-flash*	2,084	2.11 seconds	1,457.00%
text/rtf*	44,992	1.16 seconds	118.51%
Average Across All	**1,198,141**	**2.39 seconds**	**306.56%**

Table 2: Analysis Time and Data Expansion by MIME Type
*(Note that * indicates a wildcard)*

Threat Intelligence

Coupled with DFI, InQuest Labs publishes threat intelligence updates on a regular weekly or as-needed basis. These updates include IP, domain, and SSL certificate Indicators of Compromise (IoC), as well as additions/modifications/removals to the InQuest threat-detection and data-loss signature knowledge base. We regularly update the InQuest scoring algorithm, the component responsible for producing an all-encompassing threat or data-loss score (ranging from 0 through 10), in consideration of all available inputs, which may include optional integrations with reputation feeds, InQuest threat exchange, multi-av providers, and sandbox solutions. The specifics of this scoring algorithm are outside the scope of this paper. Updates to the InQuest heuristic extractors and machine learning models are also regularly published.

The InQuest Labs team sources threat intelligence through a variety of methods and origins, including proprietary harvesting methods, commercial feeds and partnerships, unique partnerships, and open source intelligence (OSINT). Noteworthy within the scope of this paper is our membership in the Microsoft Active Protections Program[6] (MAPP) and our 0day exchange with Exodus Intelligence[7]. Through MAPP, InQuest customers receive thorough and accurate protections on the day of patch release (*n*day) for all Microsoft and Adobe products. Additionally, MAPP provides an ecosystem for threat indicator exchange with which InQuest

[6] https://www.microsoft.com/en-us/msrc/mapp
[7] https://www.exodusintel.com

Labs actively engages. Through Exodus, InQuest customers receive protection against 0day vulnerabilities and bleeding edge exploitation techniques.

Intelligent Orchestration

In addition to onboard native analysis capabilities and cloud-based reputation services, the InQuest platform provides optional turnkey integrations with a variety of complementary technologies. These data flow paths are depicted in Figure 3 (page 6) as light-grey optional paths and include a variety of prominent sandboxing and antivirus consensus solutions. We define "intelligent" orchestration here as the ability to both provide data-to and ingest results-from these integrations. Ingested results are interpreted by the InQuest scoring algorithm which is updated regularly. For example, InQuest Labs provides weights and filters applied atop of antivirus results. These updates are driven by empirical observations drawn from the InQuest Labs team's daily R&D efforts and mass ingestion of malware. This analytics layer is enabled by default but can be toggled off by the customer. While further specifics regarding these integrations are outside the scope of this paper, the list of integrations is enumerated here:

- Reputation
    - InQuest file, IP, and domain reputation.
    - InQuest Threat Exchange.
- SIEM (API/CEF/syslog)
    - AlienVault
    - ArcSight
    - ELK
    - LogRhythm
    - QRadar
    - Splunk
- Anti-Virus Consensus
    - OPSWAT
    - VirusTotal
- Sandbox Solutions
    - CrowdStrike
    - Cuckoo
    - FireEye
    - Joe Sandbox
    - Palo Alto Wildfire
    - VMRay

Having set the stage with regards to data capture, retention, exposure, and analytics; we are now ready to dive into the pinnacle capability of the InQuest platform, RetroHunting.

RetroHunting: Retrospective Analysis

Hindsight is 20/20. We know we are not as smart today as we will be tomorrow. There are countless adages along these lines and equally countless benefits for defenders who are empowered with such a capability. A core mantra at InQuest is to utilize all available resources in the process of identifying malicious logic and sensitive data in-use, in-motion, and at-rest. RetroHunting is used to discover something suspicious that intel didn't identify at the initial time of analysis. We like to say that we "throw everything and the kitchen sink" at the challenge. Figures 5 and 6 below illustrate this concept.

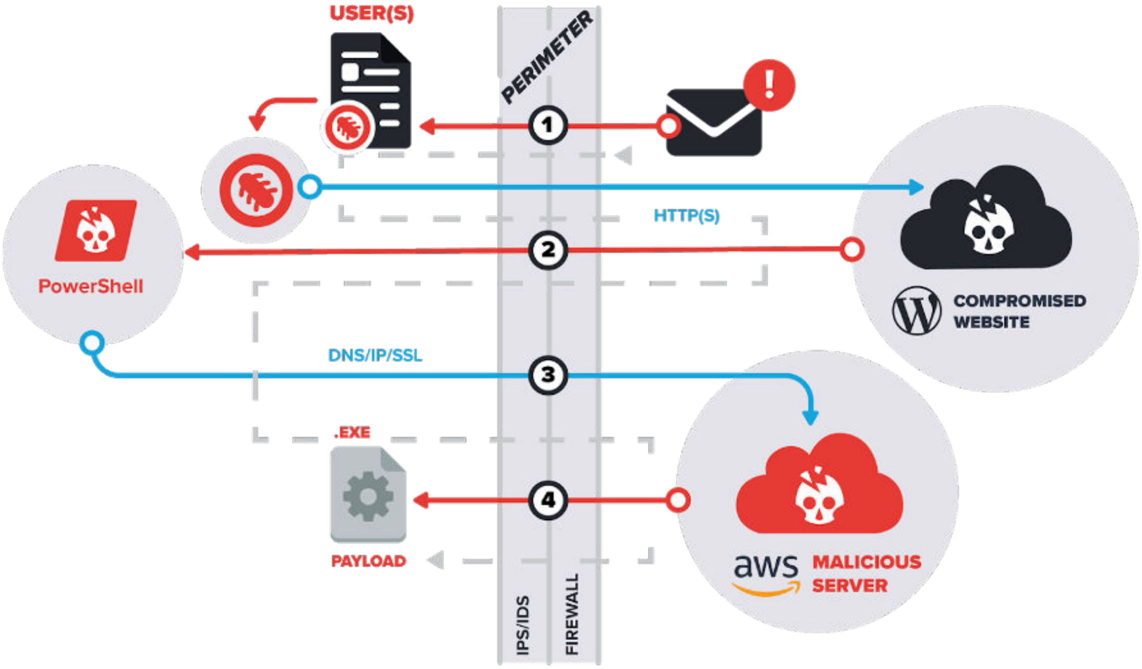


Figure 5: Common Staged Malware Campaign

Most modern-day malware operates in a series of stages. A very common pattern is depicted in Figure 5. Let's walk through the four labeled stages and align them with Figure 6, the InQuest defense-in-depth strategy.

1. Initial delivery of malware lure over email. This is often as an attachment or malicious URL embedded in the email body. Depending on the deployment methodology, InQuest can prevent or detect this delivery before the target user ever receives the payload.
2. Once activated, the malware lure typically calls out to the cloud to retrieve the next stage payload. Again, depending on deployment methodology, InQuest can prevent or detect this callback by anchoring on the communication patterns within the HTTP(S) request. For example, it is commonplace for attackers to compromise a vulnerable WordPress site to host components of their malware. These otherwise legitimate sites carry a benign reputation that can assist in bypassing detection stacks.

3. Upon successful pivot to the next stage, it is common for malware to reach out to Command and Control (C&C or C2) endpoints for further instruction. InQuest Labs tracks threat actor infrastructure and provides regular hi-fidelity updates to a list of known bad IPs, domain names, and SSL certificates. As an example, the Turla nation-state campaign leveraged a novel satellite communications tactic for data exfiltration[8], but was still anchorable due to reuse of a previously identified SSL certificate.
4. Once a Command and Control channel has been established, a targeted payload can be delivered with the intent to exfiltrate data. In some cases, custom lightweight data trickles may occur over protocols such as DNS or ICMP. In others, actors hide in plain sight through exfiltration via common file formats transmitted over standard protocols like web or email. In these cases, InQuest DFI can expose and identify transmission of sensitive data.

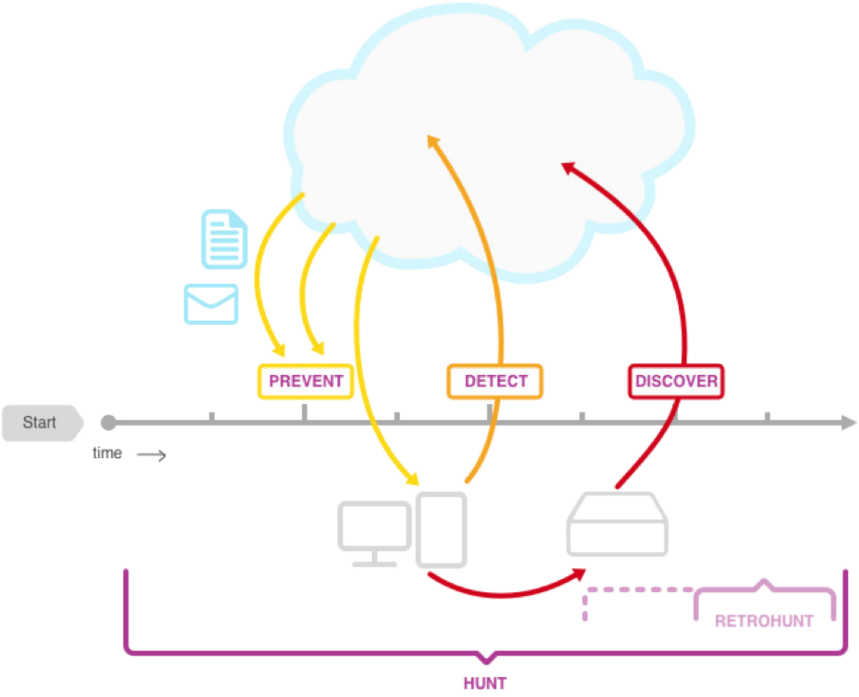


Figure 6: InQuest: Defense in Depth Strategy

In the end, there is no silver bullet, so to speak, and it is expected that a skilled and persistent adversary will breach its target network given enough time. The goal for advanced defenders in such cases is to minimize dwell time. One methodology of doing so is through threat hunting, defined as a proactive process whereby the SOC analysts search through networks, endpoints, and data in an effort to isolate and detect advanced threats that evade existing security solutions. A variety of threat hunting methods exist, some real-time and others retrospective. We'll focus on retrospection.

[8] https://media.kaspersky.com/pdf/SatTurla_Solution_Paper.pdf

Commonly, retrospective analysis is conducted over logs collected from a variety of sources. From a network perspective, retrospection has been traditionally enabled through full PCAP solutions that operate as a network flight recorder, storing every observed bit traversing the wire. These solutions are both costly and resource intensive. While a full PCAP solution provides value in a forensic capacity, it lacks in the details desired by threat hunters. This is due to the fact that captured network streams require further data processing to expose the intricate layers that threat hunters are looking for, such as embedded logic, semantics, and metadata.

RetroHunting: Timeless Deep File Inspection

The InQuest platform provides searchable access to both session and file data. Session data includes network session information, headers, and metadata. File data includes both original files and all content derived as a result of the DFI process.

Session metadata and headers are searchable directly from the InQuest UI by way of quick and advanced searches. Advanced searches allow analysts to build complex boolean searches across a variety of fields. These searches can be saved for later recall and optionally shared with other analysts (users) on the platform.

File content, both original and that derived through the DFI process, is also searchable. Analysts write YARA[9] compatible rules which combine strings, bytes patterns, and regular expressions via flexible conditional logic. These rules are saved for later recall, are editable/versionable, and can be optionally shared with other users of the platform. These written rules propagate into the InQuest knowledge base, along with the regularly updated rules from InQuest Labs. As we'll see in the next section, RetroHunting is triggered both manually and automatically.

RetroHunt Triggers and Use Cases

The retrospective analysis capability of the platform is triggered in two ways, manual and automated. Manual RetroHunts are initiated by platform users such as SOC analysts or threat hunters. Automated RetroHunts are initiated whenever new threat intelligence is added to the platform; either by InQuest Labs via our regular threat intelligence updates, or by platform users through the addition or import of user-defined signatures. The lookback window for automatic RetroHunting can be tuned by customers or disabled entirely, but by default, it is set to two weeks. RetroHunt alerts are displayed in a dedicated section of the InQuest UI/UX, have the same search interface as other data panels, and are shown in session/file details views directly below the list of real-time alerts. A countless number of use cases exist; what follows is a popular sampling of use cases.

[9] http://virustotal.github.io/yara/

Zero-Day Exploits

A zero-day (0day) is defined as a vulnerability or exploit that was discovered in-the-wild and for which no patch exists. While mitigating factors may reduce your attack surface and subsequent exposure to attack, these threats have wide reaching impact and are difficult to defend against. Some recent examples of 0day vulnerabilities having been discovered in the wild include:

- CVE-2018-4878[10], Adobe Flash DRM UAF vulnerability.
- CVE-2018-8174[11], Microsoft IE VBScript UAF vulnerability.

When such campaigns are discovered in-the-wild, InQuest Labs makes a concerted effort to both analyze the vulnerability and capture sample exploits from the field to write both generic and specific rules to mitigate the threat. Once released to customers by way of a threat intelligence update, an automated RetroHunt is triggered and will reveal if the InQuest customer was also targeted by that 0day threat. If so, RetroHunt events would be generated and automatically sent to the SIEM. SOC analysts can augment this automated trigger with a manual trigger if they wish to peer back further than the currently defined lookback window.

Data Leakage

A variety of general patterns for sensitive and personally identifiable information are bundled with the platform. For example, Social Security numbers, classified document watermarks, financial information, and more. Forethought doesn't cover all data leakage. In such cases where sensitive information was leaked and defenders want to tie that data back to the related network stream, RetroHunting can help. A user-defined signature with the relevant leaked keywords can be added to the system. If these keywords are found anywhere in semantic, meta, or Optical Character Recognition (OCR) output layers, an alert is produced. The InQuest platform offers (default disabled) an option to catalog all captured email bodies, providing yet another layer of analytics for the identification of data leakage.

Signature Testing

Another common use case for RetroHunting is utilizing the platform to test the performance of a signature on production grade data without causing network degradation or overwhelming defenders with false positives. Candidate signatures can be added to the platform and tested against real-world captured data for consideration. Results from the test can be reviewed, signatures tweaked, then re-ran for further iteration. A turnkey solution for enterprises who wish to replicate a workflow commonly utilized by IDS/IPS signature developers, but lack the resources requisite to implement a custom solution. The InQuest Labs team implements this very pattern, among a variety of other quality assurance mechanisms.

[10] https://nvd.nist.gov/vuln/detail/CVE-2018-4878

[11] https://nvd.nist.gov/vuln/detail/CVE-2018-8174

Conclusion

In this paper we provided readers with high-level background information regarding InQuest as a company and platform. We introduced a variety of deployment methods and data flow paths, and dove into the specifics around the capability provided by both Deep File Inspection (DFI™) and RetroHunting™. We outlined some common applications for how our technology can augment the efforts of defenders and threat hunters through the democratization of advanced malware analysis skills, reduction of analyst fatigue/frustration, and availing of new analytical dimensions in both data and time. The introduction of these automations allow for the reappropriation of your organizations most precious and limited commodity, human cognition, towards the more valuable human-intensive task of hunting for the threats that go otherwise unnoticed. The goal of this shift in focus is to reduce dwell time and mitigate the effects of nefarious actors performing targeted and persistent attacks against your people and your data.

For further details, we recommend the following short list of blog entries, sorted in reverse chronological order:

2019-09, Advanced Support for the SOC Hunter[12]
2018-05, RetroHunt: Retrospective Analysis for Threat Hunters[13]
2018-04, Advanced Malware Multi-Scanning On-Premises by OPSWAT Metadefender Core[14]
2018-03, InQuest Zero-Day Coverage via Partner Exodus Intel[15]
2018-03, Defense in Depth: Detonation Technologies[16]
2018-02, An Introduction to Deep File Inspection[17]

Readers interested in interacting with a lightweight version of DFI to better grasp the plausible benefits availed from additional dimensions of data (DFI) and time (RetroHunting), are encouraged to tinker with an open data portal maintained by InQuest Labs at:

https://labs.inquest.net

It should be noted that this portal hosts known malware for download to aid defenders in their research and development efforts.

For further details or to get in touch, visit us at www.inquest.net.

[12] https://inquest.net/blog/2019/9/24/advanced-support-for-the-SOC-hunter
[13] https://inquest.net/blog/2018/05/09/retrohunting-with-inquest
[14] https://inquest.net/blog/2018/04/23/opswat
[15] https://inquest.net/blog/2018/03/23/exodus
[16] https://inquest.net/blog/2018/03/12/defense-in-depth-detonation-technologies
[17] https://inquest.net/blog/2018/02/12/deep-file-inspection