

Applying **Agentic AI** to Security Operations:

Transparent and Autonomous
AI for Faster Threat Detection,
Containment, Investigation,
and Response



Executive Summary

There's nothing more important in security operations than speed. Containing threats within minutes—not hours or days—is the new standard. It's up to us in security to leverage all the latest tools to make this possible.

To help organizations combat threats and move faster, ReliaQuest has built an AI Agent within its GreyMatter technology that empowers security teams to respond even more quickly and effectively without sacrificing visibility.

It effectively gives customers the ability to train their own self-learning AI Agent on their own environment.

Agentic AI enhances security operations with faster, more accurate, and scalable threat detection, containment, investigation, and response while making autonomous decisions and recommendations.

Agentic AI refers to autonomous artificial intelligence systems that have decision-making capabilities, functioning as independent “agents.”

These AI systems can set goals, make decisions, and take action without constant human intervention. Agentic AI is often designed to carry out specific tasks, optimize processes, or solve problems by analyzing data, learning from experience, and adapting to changing conditions.

Agentic AI differs from the more familiar Generative AI models (or large-language models [LLMs]), which might be considered passive AI systems. These models may simply execute instructions or perform narrowly defined tasks without independent initiative.

Table of Contents

- Part 1: ReliaQuest AI Agent 1
 - Core Components of the ReliaQuest AI Agent..... 1
 - Model Abstraction Layer 1
 - Proprietary Knowledge Access to Reduce Hallucinations 2
 - AI Tools..... 2
 - Some Examples of ReliaQuest-Built AI Tools for Threat Detection, Containment, Investigation, and Response 2
 - Related Incidents 2
 - Intel Enrichment..... 2
 - Remediation Tools 2
 - Summarization..... 2
 - Searching Technologies..... 2
 - AI Agents..... 3
 - Planner 3
 - Executor 3
 - Agentic Workflow 4

- Part 2: The ReliaQuest Advantage..... 6
 - GreyMatter Security Operations Platform 6
 - Designed for Transparency..... 6
 - Prompt Engineering: Built by Security Analysts, Backed by Data Scientists 6
 - Decades of Incident Response Data for Smarter AI 7
 - Reinforcement Learning from Human Feedback..... 7
 - Privacy Built-In 7
 - Universal Translator to Normalize Data Across Sources 7

- Use Case: Agentic AI for Alert Investigations 8
- Conclusion..... 9
- About ReliaQuest..... 9

Part 1: ReliaQuest AI Agent

Our AI Agent helps security teams move faster with complete transparency. We built our platform with three requirements in mind:

Optionality and Modularity to Leverage the Best-Suited AI Models.

The AI landscape is evolving rapidly, and we designed our platform with the ability to test and use the right model and easily swap them out as needed.

Transparency and Accuracy.

Hallucination has been an ongoing concern for AI models, and, in the context of security operations, accuracy is paramount. We designed the AI Agent to augment its data with external information sources to provide the most accurate answers. And in the interest of transparency, we've taken things a step further by letting you see how our AI models think, plan, and execute.

Reduce Time-Intensive Tier 1 and Tier 2 Activities.

We believe AI works best to augment security operations teams. When paired with human expertise, AI models can take away the mundane tasks of security operations and allow analysts to focus on real threats.

A/B Testing: The AI Agent supports A/B testing of different generative AI models for various tasks. This involves analyzing outcomes and refining prompts for each model to ensure optimal performance and picking the best model for specific use cases.

Besides A/B testing models, the AI Agent also enables comparative testing of prompts. The process of A/B testing outputs from both models and prompts is visible to users, facilitating feedback that helps the AI self-improve.

Core Components of the ReliaQuest AI Agent

Based on these guiding principles, the ReliaQuest team has built an AI Agent using the following core components.

Model Abstraction Layer

As AI continues to change, new models will inevitably emerge. The AI Agent uses a model-agnostic architecture, allowing seamless interaction with any commercial, proprietary, open-source, or yet-to-be-released AI models.

The AI Agent helps an analyst determine the most suitable model for any given task across the threat detection, investigation, and response lifecycle. This allows for speed and efficiency, and effectively gives security teams the ability to train their own self-learning AI Agent on their own environment.

In addition to model abstraction, GreyMatter also supports:

Outcome Analysis at Scale: The AI Agent is designed for comprehensive testing and outcome analysis, enabling the comparison of different models and prompts for the same task. To conduct the required testing at enterprise scale, ReliaQuest hosts and load balances private instances of AI models.

Confidence Level: Once the accuracy and efficiency of a model has been established, our team selects the right model for each use case at the appropriate confidence level.

Proprietary Knowledge Access to Reduce Hallucinations

One of the primary challenges with LLMs is their tendency to generate plausible sounding but incorrect information, a phenomenon known as hallucination.

To combat this issue, ReliaQuest leverages Retrieval Augmented Generation (RAG) systems in real-time to provide LLMs with relevant information on historical incident response data, real-time customer context, and alert history to generate highly relevant outputs unique to each organization.

This is achieved with the following capabilities on the right.

→ Integration with Decades of Proprietary Cyber Incident Response Data:

Our RAG system is tightly integrated with ReliaQuest's vast collection of cybersecurity data, including historical alert data, threat intelligence feeds, and customer-specific information, allowing AI models to seek additional information that may not be part of their training data.

→ Context-Aware Knowledge Graph:

When processing an alert, the AI Agent also retrieves relevant customer-specific information from our data stores based on alert context. This might include similar past incidents from a specific customer's environment; known threat actor tactics; or specific customer environment details about assets, identities, and applications.

→ Augmented Generation:

The retrieved information is then used to augment the AI Agent's knowledge, effectively constraining its responses to factual, relevant data. This process significantly reduces the likelihood of the model generating false or irrelevant information.

AI Tools

AI tools are building blocks used to accomplish specific tasks. AI tools deliver single, self-contained outcomes that can be leveraged as units in the context of a workflow or by AI agents. ReliaQuest has a growing list of AI tools that we have built and fine-tuned specifically for various TDIR tasks, a few of which are detailed below.

Some Examples of ReliaQuest-Built AI Tools for Threat Detection, Containment, Investigation, and Response



Related Incidents

This tool identifies whether an alert is duplicative of another alert or if there is a correlation between alerts.



Summarization

This tool is designed to write up alerts, investigations, and case summaries, allowing security analysts to review and share information faster.



Intel Enrichment

This tool takes the initial artifact list from alert ingestion and searches against threat intel feeds. It then sends the findings to the summarization tool.



Searching Technologies

This tool creates ad-hoc searches to retrieve additional context on artifacts, such as user, server, or IP address, that are relevant to the alert being investigated.



Remediation Tools

AI tools that can interact with security tools to execute remediation actions such as resetting user credentials, isolating hosts, blocking URLs and so on.

GreyMatter maintains AI tools in a dynamic library, in which they can self-advertise their purpose, descriptions, inputs needed, and outputs delivered without needing configuration. AI agents and other services can access the AI Tool Library and choose the most appropriate tool for each task.

AI Agents

AI agents are autonomous, self-learning systems that can perform complex tasks that require reasoning, reflection, and action. ReliaQuest has built an AI agent for security operations with two main components:

Component #1: Planner

The planner acts as the “brain” of the agentic AI system, designed to formulate a series of steps to tackle specific problems. During a security alert investigation, the planner develops a workflow to arrive at a verdict for an alert based on the nature of the alert, artifact enrichment plans, customer-specific information on their environment and alert closure history, and external threat intelligence.

Example

For an “impossible travel” alert, the planner might develop a plan to first query tools for user login information, analyze the login locations, and then decide whether to reach out to the user for verification in case it identifies impossible travel patterns such as the user logging in from different continents within a short timeframe.

The planner requires extensive access to data sources, including information on detection logic to determine what triggered the alert, customer environment information, and past alert triage knowledge. The planner then generates different prompts for various alert types and dynamically adjusts plans based on real-time information.

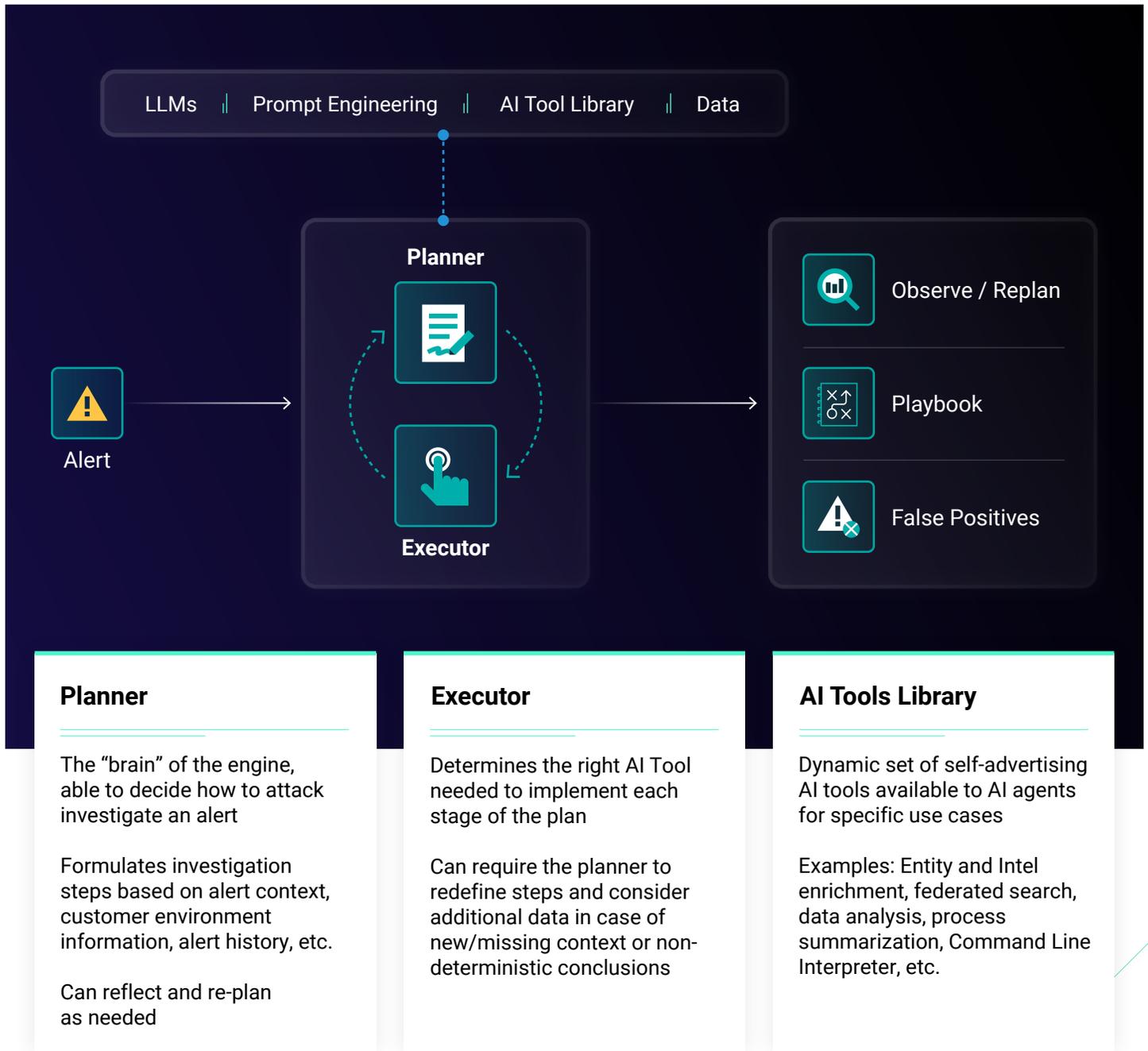
The bi-directional API and data stitching capabilities within GreyMatter allow the planner to collect data from across a customer environment to build the context needed for the investigation.

Component #2: Executor

The executor is responsible for implementing the plan devised by the planner, utilizing the appropriate tools for each step. The executor is also smart enough to observe and request a new plan if a tool fails to return data, if it is unable to reach a verdict, or if new information arises, ensuring that the overall process continues smoothly.

The executor must be built to execute asynchronously and at-scale across thousands of simultaneous plans and be able to automatically replan around failures, including those that may occur from missing or unavailable data.

Agentic Workflow



Planner

Builds the right plan based on alert artifacts, customer environment information, and past alert triage. It creates a series of steps, which may include calling tools, reflecting on tool outputs, and making decisions based on the gathered data.



Tool Selection

The planner chooses tools based on the task requirements. For example, if the task involves detecting impossible travel, the planner might:

- Use an external search tool to pull login data.
- Analyze the login data to identify unusual patterns.
- Reach out to the user for verification (e.g., confirming if they are in a specific location).



Execution

The executor implements the plan by using the selected tools. It follows the planner's instructions and may call back to the planner for replanning if new information arises or if a tool fails.

ReliaQuest's agentic approach to AI for alert investigation—including a model-agnostic architecture, robust A/B testing capabilities, load-balancing for scale, and a well-defined set of AI tools and AI agents—delivers high-fidelity security outcomes in minutes.



Part 2: The ReliaQuest Advantage

Using our guiding principles and the core components detailed in part one, ReliaQuest has built an AI system that is resilient, efficient, and capable of delivering accurate and reliable outcomes with or without human intervention.

ReliaQuest is uniquely positioned to deliver AI-driven security operations for the following reasons:

GreyMatter Security Operations Platform

AI agents are only as good as the data and context they have access to. The GreyMatter security operations platform, with its ability to integrate across 150+ endpoints, clouds, networks, and business applications gives our AI models unprecedented access to real time customer context.

This real-time data-stitching ability results in comprehensive and consistent investigations. When enabled by customers, our AI Agent can also take containment and remediation actions using the bi-directional API capabilities of GreyMatter.

Designed for Transparency

To help customers develop trust in AI, ReliaQuest provides complete transparency into the way our AI Agent operates.

Every step of a plan developed by the AI Agent—including data retrieved, inputs required, executed actions, failed steps, and observations—is displayed within GreyMatter as an internal monologue.

This transparency allows users to interrogate and provide feedback on the AI's processes, fostering trust while facilitating improvements.

The screenshot displays the AI Agent interface with a workflow on the left and a detailed plan on the right. The workflow includes steps: Generate Plan, Intel Enrichment, Replan, and Searching Technologies. The detailed plan shows a 'Generate Plan' window with a 'Result' tab containing several bullet points of instructions for the AI agent. Callouts highlight 'Autonomous Agent' capabilities and 'Human-validated feedback' options.

Autonomous Agent:

- Generates plans
- Collects additional insights
- Replans as needed

Human-validated feedback

Transparent look into the mind of AI

Generate Plan

Summary Input Parameters Result

- Use the Enriching artifacts tool to retrieve additional information about the IP address (128.136.211.22) and the domain (reliaquest.com) from the triggering artifacts to assess their reputation and any known malicious activities associated with them.
- Use the analyzing related incidents tool with the incidentId (RQ20123903) to identify any historical incidents that are similar to the current alert, which could indicate a pattern of behavior or a recurring threat actor.
- Use the command line analysis tool with the incidentId (RQ20122902) to explain the command line script or sequence of command lines present in the triggering artifacts, if any are provided.
- Use the Search Technologies tool with the incidentId (RQ20122902) and a query to retrieve data from customer technologies regarding the creation of new access tokens, specifically looking for events related to the user 'jsmith' and the session_id '2fd881cbd1ecc055288e64bed4248d6e'.
- Analyze the results from the previous steps to determine if the creation of the access token is legitimate (e.g., part of the normal user behavior or a scheduled task) or if it is indicative of malicious activity (e.g., an attacker creating tokens for persistence).

Prompt Engineering: Built by Security Analysts, Backed by Data Scientists

Prompt engineering involves crafting specific instructions for AI agents to follow to ensure they produce the desired output. Good prompting requires expertise and precision. It is essential to craft advanced and detailed prompts, specific to the capabilities of the LLM, to guide the model towards the desired outcome.

For instance, a simple prompt like "How many letters in the word strawberry are 'r'?", now infamous online, yields an incorrect answer. However, a more detailed prompt that suggests ways to verify a response, such as "How many letters in the word 'strawberry' are 'r'? **Verify** your answer by checking each letter and then count the number of occurrences of each letter," can lead to a correct answer.



ReliaQuest also incorporates proprietary prompting principles and techniques to ensure output from AI models undergoes extensive verification to generate higher-fidelity outcomes.

By employing advanced techniques such as Mixture of Prompts, Tree of Thought, Chain of Thought, and Graph of Thought, we've enhanced the quality of prompts and outputs. These techniques enable the AI agents to handle complex tasks more effectively, ensure logical progression and coherence, and provide comprehensive and accurate responses.

Decades of Incident Response Data for Smarter AI

An agentic AI outcome will only be as good as the data the AI Agents have available to them. ReliaQuest pairs decades of historical alerts and incident response data with an extensive Cyber Analysis Methodology that includes detailed information on the various alert types, a comprehensive list of steps required to fully enrich and investigate an alert, and, most importantly, the steps required to contain and resolve alerts. This information informs our prompt engineering and AI model training, allowing us to provide high-fidelity AI tools and agents.

Reinforcement Learning from Human Feedback

Incorporating human feedback is crucial for improving AI performance. ReliaQuest has a built-in feedback workflow for security analysts with domain knowledge to evaluate the output and steps taken by AI agents as a part of incident analysis. This evaluation process, paired with real data and alerts, helps refine the AI Agent's performance.

Privacy Built-In

The AI Agent is built with data privacy and security in mind. Customer data is fully segmented, and each AI execution is context-specific, operating within the boundaries of the customer's data environment. In addition, by using privately hosted AI models and data loss prevention capabilities, we ensure that proprietary or sensitive data is not exposed to public AI models or to other customer environments.

Universal Translator to Normalize Data Across Sources

For AI agents to be successful, system data must be normalized to a consistent ontology to allow for consistent, apples-to-apples comparisons. Security data from across all tools must be normalized to allow for scalable use for AI.

As part of the GreyMatter security operations platform, the Universal Translator normalizes data from multiple systems, including endpoints, SIEMs, and multiple clouds, enabling AI to process data and structure queries without depending on product-specific nuances. Because of this efficiency, ReliaQuest is well placed to scale its AI as it takes on new customers and their complex environments.

Use Case: Agentic AI for Alert Investigations

This section highlights the step-by-step thought process of an AI agent designed to autonomously investigate security alerts. In this example, the AI agent generates a plan to investigate an incident involving Nmap execution. Nmap is a common tool used by threat actors to scan an organization's infrastructure to expose vulnerabilities they can exploit. The plan lays out a triage plan using AI tools available within GreyMatter that the executor can then act on. The example also showcases the output generated by the command line AI tool.

Incident Summary

This incident involving Rule ID 787 detects the execution of Nmap, a network reconnaissance tool, which was run by user "RELIAQUEST\sa-chewell" on the host "AWUE-NP-DE01" to scan network ports on the range 172.28-29.108.62. Nmap is recognized as a utility commonly used for post-exploitation activities in cybersecurity incidents.



Triage Plan (STEP 1)

Use the "Command line analysis" tool with the provided incidentId RQ20122203 to understand the context and purpose of the command line script that was executed.

Use the "Enriching artifacts" tool with the file hash f2496b4588c96a17aba7a6a9ac06dbaf69363dd8f2b4d008555aacc40867f78f to determine if the file associated with nmap.exe is known to be malicious or if it is a legitimate version of the tool.

Use the "Analyzing related incidents" tool with the incidentId RQ20122203 to find any historical incidents that are similar to the current alert, which could indicate a pattern of behavior or provide context for the usage of the tool.

Use the "IP Analysis" tool to examine the IP range 172.28-29.108.62.

Use the "Search technologies" tool with the incidentId RQ20122203 and a query to retrieve logs from the src_host AWUE-NP-DE01 for the time around the timestamp 2024-09-13T14:40:15.062Z to see if there are any other related events or commands executed by the user RELIAQUEST\sa-chewell.

Use the "Analyzing events" tool with the queryBatchUuid and index obtained from step 3 to further investigate the events, focusing on the command line activity, especially related to the output from the command line analysis tool, which analyzed the command flagged in the triggering artifacts.

If there are any other commands that look suspicious found from the "Analyzing events" tool, use the "Command line analysis" tool to analyse what the purpose and context of these commands were.

Use the "User lookup" tool to validate that the user's command line activity aligns with their job role.

Review the results from the previous steps to determine if the usage of nmap.exe was part of legitimate administrator activity or if it is indicative of malicious behavior. If the usage is in line with normal administrative tasks and there are no other suspicious indicators, it may be considered a false positive. If the usage is unusual or there are additional indicators of compromise, further investigation will be required.

Based on the findings, conclude whether the security incident is a false positive or a true positive and recommend any necessary response actions.



Command Line (STEP 2)

The command is command line for nmap, which is a popular network scanning tool used for network discovery and security auditing. Here's a breakdown of the command and its options:

`nmap`: This is the command to run the nmap network scanner.

`--reason`: This option tells nmap to provide the reason why a specific port is set to a particular state (e.g., "reset" for closed ports).

`--open`: This tells nmap to only show open ports. Closed or filtered ports will not be displayed in the output.

`-Pn`: This option skips the discovery phase, where nmap tries to determine if the host is online before scanning it. It treats all hosts as if they are online, which is useful when scanning hosts that are blocking ping probes.

`-p 22,88,389,636,464,443,3389,8000,8443,8051,8089,8501,139`: This specifies the list of TCP ports to scan. The ports listed are commonly used for services such as SSH (22), Kerberos (88), LDAP (389), LDAPS (636), Kerberos change/set password (464), HTTPS (443), Remote Desktop Protocol (3389), and various other services that might be running on the additional ports listed.

`172.28-29.108.62`: This is the target specification for the scan. It indicates that nmap should scan the range of IP addresses from 172.28.108.62 to 172.29.108.62.

In summary, the command is instructing nmap to scan a range of IP addresses for open ports among a specified list of common service ports, without attempting to ping the hosts first, and to report the reason why each port is in a particular state. This type of scan is typically used to identify potential points of entry on a network and to understand the services that are exposed to the network, which is a critical step in a security assessment.

Conclusion

ReliaQuest has harnessed decades of SecOps data to train generative AI and agentic AI models within its GreyMatter platform, making it uniquely suited for customers looking to augment their SecOps teams. Pairing these AI capabilities with automation speeds threat detection, containment, investigation, and response even further, resulting in mean times to contain (MTTC) of 5 minutes or less for our customers.

About ReliaQuest

ReliaQuest exists to Make Security Possible. Our security operations platform, GreyMatter, allows security teams to detect, contain, investigate and respond to threats within minutes, regardless of the data source. GreyMatter eliminates Tier 1 and Tier 2 security operations, allowing security teams to focus on what matters most. ReliaQuest is the only cybersecurity technology company that delivers outcomes specific to each organization's unique architecture, technology and business needs—making the customer the platform.

With over 1,000 customers and 1,200 teammates across six global operating centers, ReliaQuest Makes Security Possible for the most trusted enterprise brands in the world. **Learn more at www.reliaquest.com.**

