digiDations

ATLAS™ Security Validation Platform Product Guide

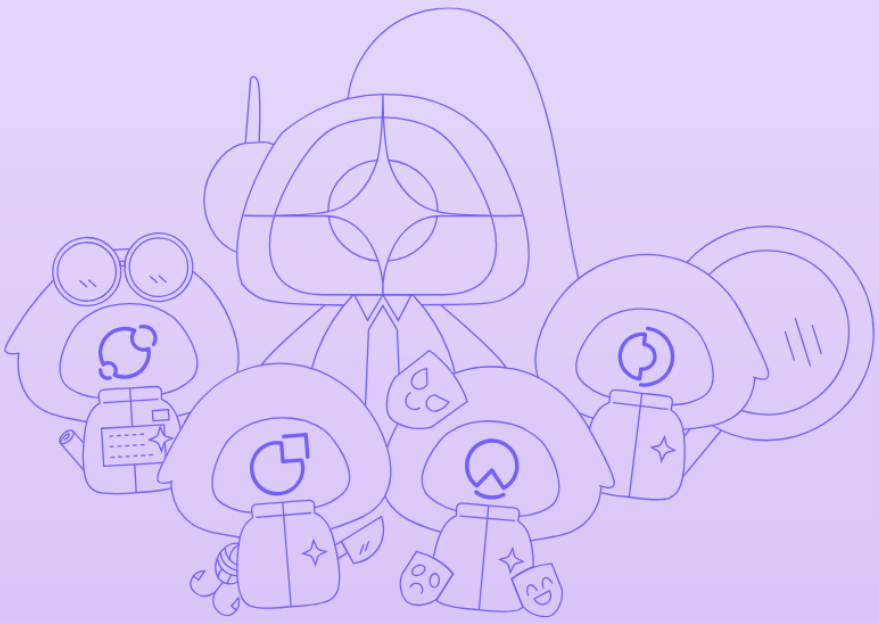


Table of Contents

Executive Summary

The ATLAS Security Validation Platform by digiDations redefines how organizations assess and maintain their security posture. Unlike static assessments or conventional Breach and Attack Simulation (BAS) tools, ATLAS enables continuous validation of security controls through real-world attack simulations that align with frameworks such as MITRE ATT&CK and the Cyber Kill Chain. Its four-stage validation journey, Verify, Evaluate, Analyze, and Optimize—provides comprehensive visibility into misconfigurations, detection gaps, correlation issues, and response readiness.

ATLAS supports flexible deployment across SaaS, on-premises, containerized, and air-gapped environments with minimal operational impact. Its core components include the ATLAS Console, Validators, Detectors, Isolators, Log Gateways, and Proxy Gateways. These modules work together to simulate adversary behavior, collect telemetry, and validate detection, response, and mitigation across complex infrastructure.

The platform incorporates advanced AI capabilities through Threat Analysis and Research Assistance (TARA AI), which enhances threat analysis, delivers remediation guidance, and generates dynamic scenarios using large language models such as Gemma and Llama. These AI integrations also support rule enrichment and threat intelligence processing to deepen simulation fidelity and improve analyst efficiency.

ATLAS is designed with strong platform safeguards, including encrypted communications, isolated execution environments for destructive testing, and robust data protection practices. Its continuous validation cycle enables organizations to establish a defensible security baseline, monitor for security drift, and enhance resilience over time.

Supported by expert threat research, real-time intelligence integration, and a scalable system architecture, ATLAS helps security teams validate more effectively, respond faster, and defend with greater confidence. It bridges the gap between perceived security and actual resilience.

The Reality of Modern Security Strategy

Over the past two decades, organizational security strategies have grown increasingly complex. What began as basic antivirus tools have evolved into layered security architectures designed to detect, prevent, and respond to a broad range of threats. Today's adversaries are sophisticated threat groups capable of executing coordinated, multi-vector attacks across on-premises, cloud, and hybrid environments.

As businesses expand globally, embrace the cloud, and support mobile workforces, their attack surface continues to grow. In response, security teams have deployed a wide array of technologies, implemented overlapping detection and prevention controls, and established formal incident response procedures. Many organizations also operate dedicated security operations centers (SOC), supported by offensive security and penetration testing teams.

Despite these investments, critical security incidents remain common. The gap between perceived security and actual resilience has led many executives to ask: Why do breaches still happen, and how ready are we if one strikes? The following sections explore the core challenges driving that uncertainty.

1.1 Security Based on Assumptions

Many security strategies remain built on unverified assumptions:

- Believing that security products perform as vendors claim
- Trusting those tools are properly deployed and configured according to best practices
- Expecting security teams to consistently follow established incident response processes
- Assuming personnel fully understand infrastructure changes and apply them correctly

1.2 Critical Security Questions with Unclear Answers

- What is our current security posture, and how do we validate its effectiveness with confidence?
- If a known threat group targeted us using their latest techniques, could our defenses detect and stop them?
- If an attack like the one on Company A happened to us, could we prevent or contain it?

- Where should we increase or reduce our security investments, and what data supports those decisions?
- How do we prove that our defenses are stronger than they were six months ago?

1.3 Limitations of Traditional Penetration Testing

Despite being a widely used security assessment method, traditional penetration testing faces several well-known limitations:

- Tester skill levels vary significantly, affecting the depth and realism of assessments.
- Short engagement windows make it difficult to maintain continuous visibility or assurance.
- The range of techniques used is often narrow due to time constraints and personnel expertise.
- Testing on production systems requires guardrails around methods and malware to minimize damage. This approach reduces the test's realism and depth.
- The primary focus is on vulnerability discovery, not validating whether the full defense stack can detect or stop real-world attacks.

1.4 Limitations of Traditional Red Teaming

While red team exercises provide valuable insight into organizational defenses, they also face several key limitations:

- Offensive team skill levels vary widely, affecting the scope and realism of the exercise
- Exercises are typically conducted once a quarter or annually, limiting their value for continuous assessment
- The breadth of techniques used is constrained by time and expertise
- In simulations, blue teams often react faster, but real attacks bring confusion, stress, and tighter resource limits
- Offensive teams avoid using destructive techniques or real malware, which prevents full validation of defensive tool effectiveness

1.5 The Case for Security Validation

In today's security environment, no matter how advanced or well-funded an organization's strategy may be, it is no longer sufficient to rely solely on defensive experience and internal

assumptions. Security teams must evaluate their defenses from the attacker's perspective to uncover blind spots and weaknesses that traditional approaches may overlook.

This growing need has led to the rise of **security validation** solutions purpose-built to test an organization's ability to detect, prevent, and respond using real attack techniques. By continuously validating the effectiveness of the entire security posture, these solutions provide a deeper understanding of both adversary behavior and defensive readiness. The result is improved effectiveness, faster adaptation, and stronger overall resilience.

Four Stages of Security Validation Journey

While many solutions aim to assess an organization's true security posture, they often share common limitations. Testing is frequently treated as a periodic activity, for example quarterly penetration tests. Traditional solutions often focus solely on individual controls and their ability to detect or block specific threats.

digiDations takes a broader, continuous approach with a four-stage validation journey to deliver the most accurate and up-to-date view of your security posture. This framework not only tests defenses but also verifies baseline configurations, confirms the ability to identify incidents, and assesses response processes. By embedding these stages into ongoing workflows, ATLAS fills the gaps left by traditional solutions and provides measurable, data-driven confidence at every step.

1. **VERIFY (Operational Design)**
   Validate your security configurations and baseline controls to maintain a resilient defense foundation. In this stage, ATLAS executes basic attack techniques to confirm that each security control responds as expected and to uncover any misconfigurations before advancing to more complex testing.
2. **EVALUATE (Security Effectiveness)**
   Validate your security controls against real-world attack scenarios to maximize your protection investment. ATLAS draws on the latest techniques and threat intelligence from its comprehensive attack library to simulate adversary behavior, ensuring every control can detect and block threats at the expected level of effectiveness.
3. **ANALYZE (Threat Correlation & SecOps Efficiency)**
   Validate your correlation rules through realistic attack sequences to optimize detection accuracy and reduce alert fatigue. ATLAS leverages multi-stage scenarios and playbooks that mirror actual kill chains, potentially engaging multiple controls to confirm that your SIEM and analytics pipelines correctly aggregate and prioritize events into meaningful incidents.

4. **OPTIMIZE (Incident Response Readiness)**
   Validate your end-to-end incident response workflow to ensure timely and effective threat mitigation. From initial detection through ticket creation and containment, ATLAS measures true mean-time-to-detect (MTTD) and mean-time-to-respond (MTTR). Detailed timestamps for each event—such as the interval between SIEM alert and ITSM ticket opening—give you the granularity needed to fine-tune processes and improve your overall response posture.

Together these stages form a continuous validation loop—verify configurations, evaluate controls, analyze detection logic, and optimize response readiness. By embedding this journey into your regular change management and vulnerability assessment processes, you maintain confidence in your defenses and close the gap between perceived security and actual resilience.

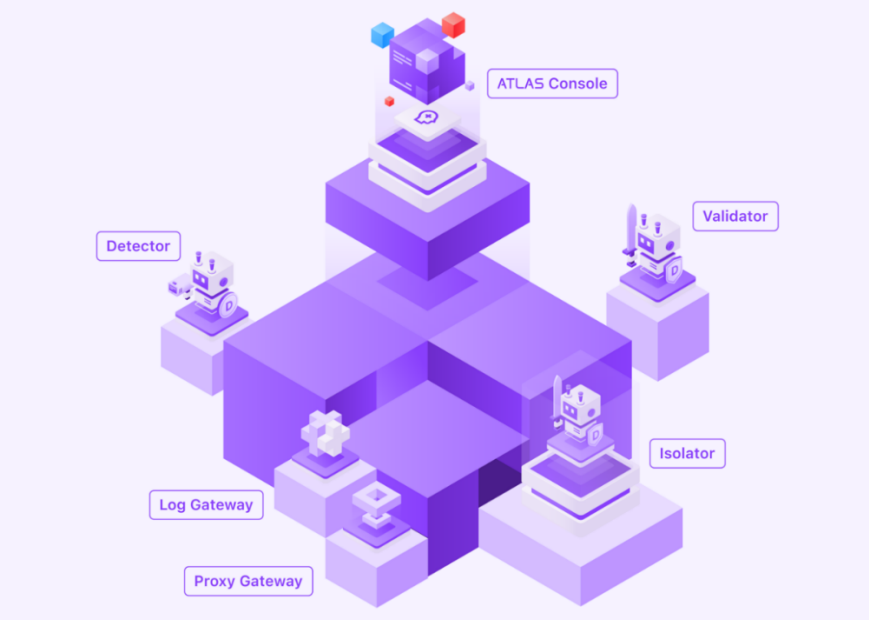
ATLAS Security Validation Platform

The digiDations **ATLAS Security Validation Platform** is a real-time validation and monitoring solution that continuously tests, measures, and verifies the effectiveness of an organization's security defenses. It uses real-world attack techniques mapped to all stages of the attack kill chain to simulate threats, uncover blind spots, and provide actionable insights on how existing tools and processes perform under pressure. These scenarios are designed to mirror actual adversary behavior, allowing security teams to understand their environment from the attacker's perspective.

This continuous validation process ensures that deployed controls are functioning as expected, proactively identifies security drift caused by environmental changes, and generates alerts when deviations occur. ATLAS also delivers data-driven insights that help security teams prioritize remediation and improve decision-making. It further supports operations training by enabling teams to practice and evaluate their incident response procedures in live environments, which ultimately improves readiness and strengthens the organization's overall defense posture.

2.1 Platform Architecture

ATLAS Security Validation Platform validates security effectiveness by deploying lightweight, flexible elements within the organization's network environment. The platform currently includes the following main modules:



ATLAS Console:

The ATLAS Console provides centralized control for simulation orchestration, validation rule

management, and threat intelligence synchronization. Users can configure and schedule simulation jobs, which are then distributed to designated modules for execution according to predefined scripts. ATLAS Console collects and aggregates logs, empowers correlation use case creation and validation, and revalidates defensive outcomes. It also provides reporting capabilities to review historical results and support operational analysis. ATLAS Console supports full WUI (Web User Interface) function API integration with other platforms to achieve automation that include but not limited to, validation library search, validation job creation, validation job result retrieval, validation rule customization etc.

Validator:

Validators execute simulated attacks based on job packages assigned by the ATLAS Console. They operate in pairs, with one acting as the source and the other as the target of a given simulation. The same Validator may serve as the source in one task and as the target in another, depending on task configuration. Simulations are non-destructive and isolated from production systems.

Detector:

Detectors are always configured as the source of an attack and, unlike Validators, target systems specified by the user. They send real attack payloads to designated hosts to observe how the target system and its associated security controls respond under conditions that are indistinguishable from an actual adversary.

Isolator:

The Isolator is a separately deployed sandbox environment that only communicates with the ATLAS Console. It executes validation rules involving destructive attack behaviors, allowing simulation of final-stage or high-impact adversary actions without affecting the production environment. All ACL (Access Control List) rules must be configured and enabled by the administrator.

Log Gateway:

The Log Gateway is a separately deployed component to collect and filter logs from security tools, including endpoint and network products, SIEM platforms and SOC systems, and forward them to ATLAS Console for validation analysis. It integrates using APIs, Syslog, or Kafka, and filters logs based on Validator IP addresses during collection. Relevant entries are forwarded to ATLAS Console for correlation and analysis. If logs are centrally aggregated in a platform such as a SIEM, Log Gateway can collect related alert logs using the same filtering logic.

Proxy Gateway:

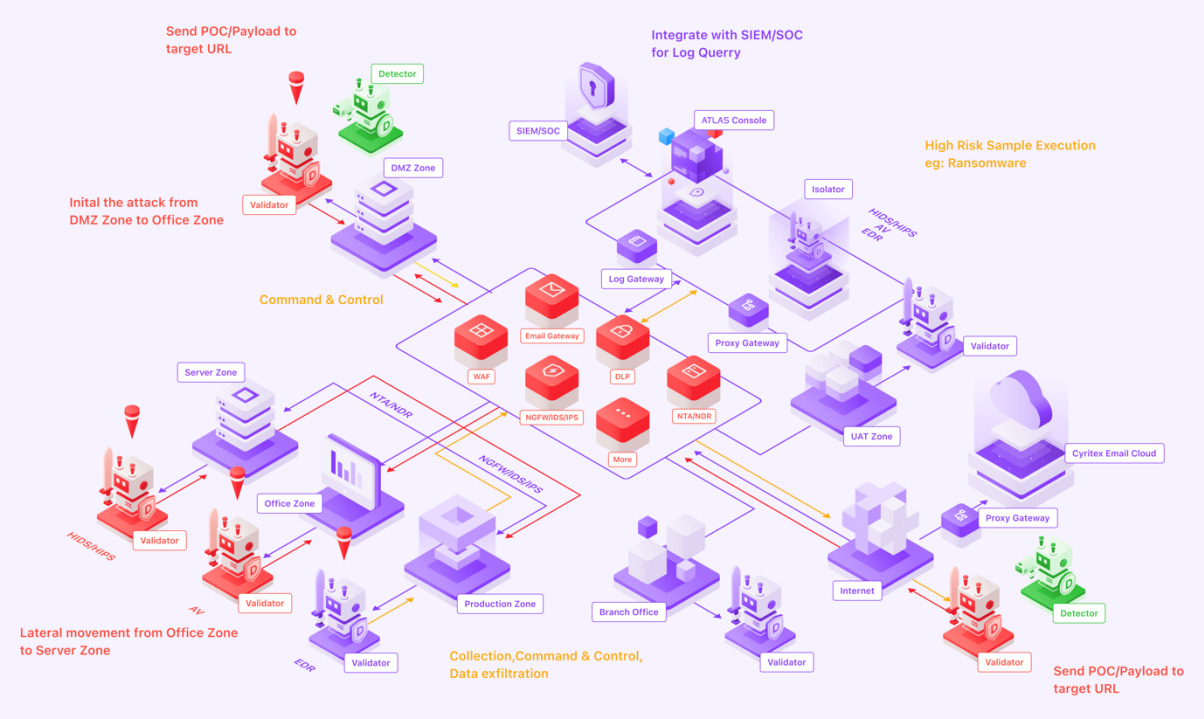
The Proxy Gateway is deployed as needed to handle communication restrictions between the ATLAS Console and other modules in complex environments. It supports both forward and reverse proxy modes to accommodate different network architectures.

WAF Sensor:

The WAF Sensor determines whether a Web Application Firewall is protecting a given URL by sending common web attack payloads, such as SQL injection and cross-site scripting, to the target. It will report the outcome to the ATLAS Console for logging and orchestration.

2.2 Deployment Architecture and Data Flow

The diagram below illustrates a typical deployment architecture. Validators are deployed across different security zones. Detectors are deployed in the DMZ (Demilitarized Zone) and externally. The ATLAS Console, located in the management network, manages all Validators, Detectors, the Isolator, and Gateways.



During simulation execution, network-based attack traffic traverses inline or bypass-mode security devices, while host-based attacks are executed directly on systems with security software installed. The ATLAS Console evaluates validation results by analyzing alert logs collected from the relevant security products. In the example shown, the attack originates from a Validator located in the Internet zone and targets a Validator acting as the target within the DMZ zone.

2.3 Attack Scenario Orchestration

ATLAS includes a validation library of over 20,000 validation rules, each with a predefined attack action or sequence and enriched metadata such as threat intelligence, technique classification, and detection criteria. These rules form the foundation of simulation jobs and are continuously updated based on evolving adversary behaviors.

To simplify rule selection and planning, the validation library content is categorized and structured across several dimensions:

Quick Start Packs

Provides preconfigured simulation packages designed for rapid deployment. Each package includes curated sets of validation rules aligned to common use cases such as testing the effectiveness of specific security controls, simulating recent high CVEs, evaluating exposure to known APT groups or ransomware families, and validating common web application attack vectors. Future additions will align more closely with the four-stage validation journey to support scenario selection by validation objective.

Playbook Builder

Constructs logical attack campaigns by chaining together validation rules with conditional logic. Each play represents a specific simulation instance aligned to a known threat actor, malware strain, or attack pattern. The logic adapts in real time, proceeding along different paths depending on whether previous steps succeed or are blocked. Plays ~~can~~ model complete or partial scenarios, allowing flexible and resilient campaign simulation.

Scenario Explorer

Group validation rules based on specific attack intents or operational focus, such as simulating malware activity (e.g., WhisperGate), testing exploitation of a known vulnerability (e.g., Log4Shell), or evaluating the performance of a particular security control (e.g., EDR, NGFW, email security). Each scenario represents a targeted objective and contains multiple rule variations to assess how well defenses perform under realistic conditions tied to that use case.

ATT&CK Map

Maps validation rules to the MITRE ATT&CK framework, allowing users to view rule coverage across tactics and techniques. Both **Enterprise** and **ICS (Industrial Control System)** matrices are supported. Users can explore validation rules grouped under each tactic, drill into individual techniques, and see how many rules exist for each mapping. This view supports structured threat coverage assessments and helps identify detection and prevention gaps based on adversary behavior.

Kill Chain Map

Aligns validation rules to the Cyber Kill Chain framework, organizing them by phase such as Reconnaissance, Delivery, Exploitation, and Command and Control. This view helps users construct full attack paths by understanding which rules test controls at each phase of the adversary lifecycle. It supports sequential testing across the chain to assess layered defenses and identify where detection or blocking may fail in a multi-stage attack.

Rule Explorer

Enables direct access to over 20,000 individual validation rules, each with defined actions and rich metadata. Users can search, filter, and investigate rules based on tactics, threat actors, attack methods, file signatures, detection tags, and more. Each rule contains detailed information, including threat intelligence context, mitigation recommendations, payload and commands, associated alert samples, and execution history, providing the transparency needed to understand how each simulation behaves and is expected to be detected.

Custom Rule Builder

Enables users to define tailored validation rules by specifying simulation types, content parameters, and execution logic. Through an interactive interface, users can select from various rule types, such as phishing emails, file transfers, or DNS queries, and configure details including payload content, network behavior, and conditional steps. The builder supports both simple and complex rule definitions, including multi-step logic, attachment upload, port targeting, and operator chaining using AND/OR conditions. This flexibility empowers organizations to replicate emerging threats, reproduce internal findings, or test specific hypotheses that are not covered in existing validation content.

This layered structure enables flexible simulation planning, supporting everything from simple rule-level validation to complex multi-stage adversary emulation, without requiring users to manually search for or configure individual actions.

2.4 Validator-Based Attack Simulation Design

In ATLAS simulations, Validators operate in coordinated pairs: one assigned as the source of the attack and the other as the target. The same Validator may serve in either role across different tasks, depending on simulation requirements. This flexible design avoids the need to bind a Validator to a fixed role.

Validators must match the appropriate operating system context for the simulated technique—for example, a Linux Validator is required for Linux-based exploits. However, they do not need to run the actual application or service being tested. A single Validator can serve as the target for a wide range of techniques, such as web application exploits or email-based

attacks, without hosting a web server or mail system. This eliminates the need to replicate complex production services and avoids exposing real vulnerabilities that adversaries could abuse during simulation.

This approach reduces the number of Validators required for deployment, simplifies access control rule configurations, and minimizes network complexity. It also enables simulations to more accurately reflect real-world adversary behavior.

During execution, the attacking Validator sends packets while the target Validator returns predefined responses. Traffic flows through network and endpoint security controls. If the attack is detected, the session is terminated and a blocked result is reported to the ATLAS Console. If undetected, the simulation completes and an unblocked result is recorded.

Platform Security Mechanisms

3.1 Security Assurance and Platform Safeguards

ATLAS is engineered to validate defenses without disrupting production systems or exposing the environment to unnecessary risk. Platform design prioritizes secure operation, realistic threat simulation, and strict control over destructive behaviors.

- All communication between platform components is encrypted, including agent traffic, log transfers, and control signaling.
- Validator, Detector, and Gateway registration uses token-based authentication with encrypted credentials issued by the ATLAS Console.
- The ATLAS Console enforces OS hardening and data encryption, ensuring secure storage of tokens, configurations, and simulation data.
- Agents are bound to a specific ATLAS Console and are not permitted to communicate with unauthorized consoles or other agents.
- All sensitive data, including registration tokens, simulation packages, and heartbeats, is encrypted in transit using secure protocols.
- Destructive or high-risk validation rules are isolated to the Isolator, a separately deployed sandbox that prevents malicious samples from affecting production systems.
- Attack simulations target Validators, not production servers or business applications, ensuring that core infrastructure remains untouched during testing.

3.2 Network Communication Security

All ATLAS platform components communicate over **TLS 1.2 via HTTPS on TCP port 443**, including traffic between the Console, Validators, Detectors, Gateways, and the Isolator. This applies to log transfers, task distribution, heartbeat signaling, validation library updates, and control messages. All communication is encrypted and adheres to enterprise firewall policies; no proprietary ports or protocols are required.

Platform components communicate only with the ATLAS Console to which they are registered. Peer-to-peer communication is not permitted, except between paired Validators when executing a simulation. Administrators access the platform through a secure web interface, also served over HTTPS on port 443.

By default, **SSH access to the ATLAS Console is disabled**. It can be manually enabled when required for controlled administrative access. Additionally, the Console connects to

digiDations' update server over HTTPS (port 443) to retrieve platform updates and threat intelligence.

This architecture ensures centralized control, minimizes the communication surface, and maintains strict segmentation between simulation infrastructure and production environments.

3.3 Agent Deployment and Registration Controls

ATLAS uses a secure registration mechanism to ensure that only authorized Validators can communicate with their assigned Console and receive job assignments. The registration process prevents unauthorized devices from being added and ensures that each Validator is cryptographically bound to a specific Console. The process is as follows:

1. An administrator creates a new Validator entry in the Console, which enters a pending state. A one-time registration code is automatically generated.
2. After the Validator is deployed, it initiates a registration request using this code.
3. The registration code is transmitted over an encrypted channel between the Validator and Console. Once verified, the Validator is moved from the pending list to the registered list.
4. Upon successful registration, the one-time code expires. A persistent token is generated and securely exchanged between the Validator and Console. All subsequent communication is authenticated using this token over TLS.

The registration token is stored in an encrypted field within the Console database to prevent tampering or misuse. Validators are not permitted to register with multiple Consoles or switch Consoles post-registration, ensuring strict binding and integrity.

3.4 Handling of Malicious Samples

ATLAS uses real-world attacker TTPs and malicious samples to ensure authenticity and realism in simulation. This approach allows organizations to assess the detection fidelity of their security controls and evaluate potential false positives. To protect production environments, the platform incorporates:

1. **Controlled Communication**
   Validators only communicate with one another when executing a simulation job coordinated by the Console. Unsolicited communication is not permitted.
2. **Secure Network Simulation**
   For network-based simulations, the Console controls both the source and target Validators. These Validators simulate specific system types to maintain realism, using

packet replay or controlled behavior. There is no need to run real vulnerable services or applications, so the simulation does not introduce additional risk to the underlying system or overall environment.

3. **Rule Classification and Execution Control**
   The validation library categorizes endpoint/server validation rules into **destructive** and **non-destructive** types.
   - Non-Destructive rules can be executed directly on registered Validators in the environment.
   - Destructive rules, such as those simulating disk wipes, ransomware, or bootloader tampering, are restricted to execution within the Isolator, a sandboxed environment with snapshot and rollback protection.
   - Destructive validation rules identified by a "Sandbox" prefix can only be executed on an Isolator. When configuring these rules, the ATLAS Console enforces selection of an Isolator to prevent operational errors.
   - Any newly created destructive rule using the Custom Rule Builder must be explicitly approved by an authorized user before execution.

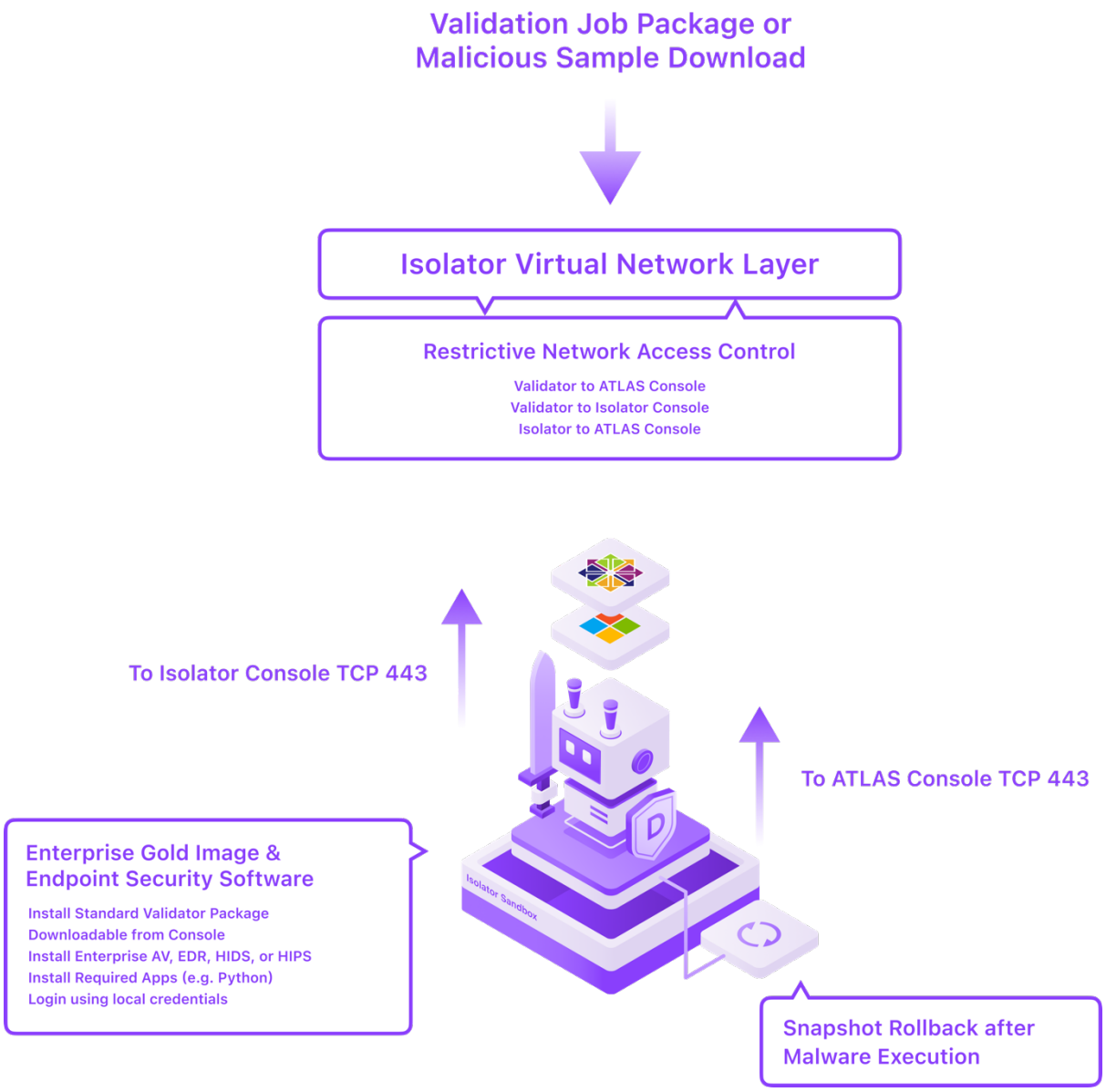
This layered control model ensures that even high-risk simulations can be safely executed without impacting business applications or production systems.

3.5 Isolated Execution of Destructive Behaviors

The Isolator provides an isolated and nested virtual environment to execute destructive validation rules safely. It runs virtualized guest operating systems using QCOW2 (QEMU Copy-On-Write version 2) image within the sandboxed environment.

1. **Importing Gold Image**
   Administrators can import enterprise-standard images (Golden Images) into the Isolator for different guest operating systems.
2. **Validator Installation**
   After importing the golden image, users can access the Guest OS through the Isolator management console to complete the installation of the Validator software after initially setting it up in the ATLAS Console. This process is the same as standard Validator install outside the Isolator. If endpoint security software is not pre-installed in the image, users can manually install it at this stage.
3. **Allow Validator Operation**
   The Validator executable and process hash must be added to the allow list of any installed security controls (e.g., antivirus, EDR, HIDS, or HIPS) to ensure normal

Validator operation. This step applies to all Validators, including those running within Isolator.



Isolator Virtual Network Layer

Isolator operates within a dedicated virtual network designed to securely execute destructive validation rules without risking external exposure. This virtual network isolates all communication pathways and enforces strict control over traffic flow, similar to a firewall. It also includes internal DNS and NTP services to support simulation behaviors without relying on external infrastructure.

1. **Customizable Allow List**
   By default, Validators inside the Isolator environment are only permitted to communicate with the ATLAS Console over TCP port 443. Administrators can modify the configuration to allow communication with additional IP addresses, hostnames, or ports—such as those required by endpoint security software management consoles.
2. **Traffic Restriction**
   All non-allowed traffic is automatically blocked by the virtual network layer, ensuring containment and control throughout the simulation process.
3. **Internal DNS Service**
   The internal DNS server resolves all queries to a default IP address (1.2.3.4), which can be modified through the ATLAS Console. If custom DNS responses are required for a simulation, they can be configured directly through the ATLAS Console.

Rollback Mechanism

The rollback mechanism in the Isolator ensures that every destructive simulation is followed by a clean recovery process. When a Validator is first installed inside the sandbox environment, the system automatically captures an initial snapshot of the Guest OS. Administrators are advised to complete all endpoint or server security software installations, policy updates, and NTP time synchronization before executing any malicious samples.

Additional snapshots can be created manually via the Isolator's management interface. The system also performs automated daily snapshots, retaining the three most recent versions.

After the execution of any malicious sample or validation rule involving destructive behavior, the Guest OS is automatically reverted to the most recent snapshot. No additional simulation activity is allowed until this rollback is completed and the Validator re-establishes communication with the ATLAS Console. This process guarantees a known-good baseline and prevents persistent contamination, allowing safe and repeated testing of high-risk scenarios.

AI Integration and Architecture

4.1 AI-Powered Analysis and Remediation Support

TARA AI (Threat Analysis and Research Assistant) is integrated into the ATLAS Console and supports users across multiple stages of the validation lifecycle. It is designed to help analysts interpret threat intelligence, understand validation content, and act on findings with greater confidence. TARA is powered by a trained large language model (LLM) deployed within digiDations' secure cloud environment, enabling advanced analysis while maintaining strict control over user data and system access.

TARA can investigate validation rules, analyze threat behaviors, and explore potential responses to evolving adversary tactics. Key capabilities include:

1. **Threat Behavior and Payload Analysis**
    - Describes commands, payloads, and behaviors observed in the simulation
    - Highlights potential risks based on emulated attacker intent and technique
    - Helps teams to understand how the simulated behavior would impact real assets
2. **Detection Rule Recommendations**
    - Suggests detection logic compatible with various security tools and frameworks
    - Supports rule formats including Suricata, ModSecurity, Sigma, and regular expression
    - Supports customization and direct integration into security product configurations
3. **Mitigation Guidance**
    - Recommends remediation steps to address misconfigurations or detection gaps
    - Aligns actions with the specific threat modeled in the validation rule or scenario
    - Helps security teams prioritize fixes based on severity and likelihood of exploitation

With TARA AI embedded directly within the ATLAS Console, users can now transform complex threat analysis into immediate, actionable steps. Instead of spending days manually dissecting payloads, researching potential risks, and updating detection policies across disparate systems. Security teams can generate detection rules, evaluate mitigation options, and initiate retesting, all within minutes. This AI-powered integration streamlines investigation,

accelerates response, and reinforces the value of ATLAS as a continuously validating, continuously optimizing security platform.

4.2 Report Generation via TARA AI

When a validation task completes based on the user's orchestration or manual selection, the platform allows users to export the relevant reports at any time. If users are unfamiliar with the console interface or prefer a guided workflow, they can rely on **TARA AI**, the embedded assistant, for help.

By simply entering a natural-language request such as **“Export report”** into TARA AI's chat interface, the assistant will interpret the user's intent and prompt for the necessary parameters to generate the desired report. TARA AI can assist users in selecting:

Report Types:

- ATT&CK Matrix Report
- Kill Chain Stage Report
- Threat Group Report
- Endpoint Security Product Report
- Network Security Product Report
- Email Security Product Report
- Security Operations Center Report
- Vulnerability Report
- Malware Report
- Ransomware Report
- Detection Report
- WAF Coverage Report
- Basic Defense Integrity Validation Report
- Defense-in-Depth Capability Assessment Report

Report Filters (Optional):

- Start Time
- End Time
- Security Zone
- Validator
- Source Validator
- Target Validator
- Job ID
- Playbook ID

Output Formats:

- PDF
- Word
- HTML

TARA AI understands context-specific input, prompts for missing information, and guides users through task completion. All prompts and outputs are scoped to platform-relevant content only. digiDations ensures that the assistant’s responses are strictly constrained to supported features, maintaining data integrity and compliance throughout its use of AI-driven capabilities.

4.3 Dynamic Attack Simulation and Rule Enrichment

AI-Powered Rule Enrichment

As adversaries increasingly use large language models (LLMs) to generate payload variants, exploit vulnerabilities, and craft phishing content, defenders must validate their detection capabilities against these evolving, AI-generated threats. To support this, TARA AI can generate intelligent variations based on existing attack rules—enabling faster rule enrichment without waiting for library updates.

Users can select any rule in the validation library involving payloads or files (such as web access or file transfer). In addition to standard configuration parameters, users can enable the **"AI Derivation"** option and specify the desired number of variants. TARA AI will then generate equivalent rules with the same intent but modified characteristics, allowing organizations to simulate and validate detection coverage across a broader range of threats.

Phishing-related rules can also be enriched in this way. Users may select an existing phishing email rule and choose either to:

- Automatically generate variants based on the original email content, or
- Provide one or more keywords, which TARA AI uses to create new phishing emails and rules.

This capability ensures that the validation library remains dynamically extensible and reflective of current threat evolution.

Scenario and Playbook Generation via TARA AI

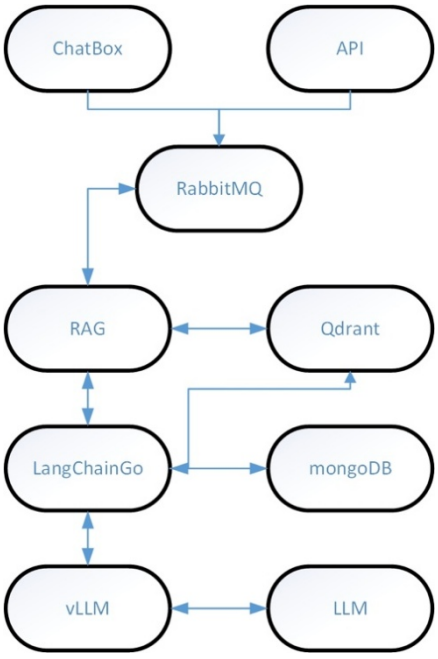
Some users may not interact with the platform regularly or may be unfamiliar with designing effective simulations. TARA AI provides direct assistance by generating relevant scenarios or playbooks using natural language input.

Users simply describe the desired simulation context using the TARA AI chat interface—this can include tactics, techniques, and sub-techniques from the MITRE ATT&CK Framework or stages of the Lockheed Martin Cyber Kill Chain. TARA AI then generates a tailored scenario or playbook and prompts the user for execution parameters, including:

- Security Zone
- Source Validator
- Target Validator
- Operating System
- User Permissions
- Domain User Binding
- Communication Protocols
- File Deployment Path
- Security Products
- Execution Time

Once essential parameters are provided, the job is automatically assembled and ready for execution. TARA AI can also retrieve and re-run historical jobs based on ID or description, eliminating the need for manual search and reconfiguration.

4.4 AI Engine Architecture and Model Support



The AI system within the ATLAS Security Validation Platform is built using a modular architecture that includes TARA AI (accessed via the ChatBox interface), an API layer, RabbitMQ, Retrieval-Augmented Generation (RAG), Qdrant, LangChainGo, MongoDB, vLLM, and a connected large language model (LLM). This architecture ensures secure and compliant use of LLM technologies while delivering high performance and responsiveness to end users.

To minimize system load and preserve GPU resources, the platform employs multiple optimization techniques such as effective prompt engineering, dynamic load balancing, and the use of a local knowledge base. These measures reduce unnecessary LLM queries while maintaining high output quality and consistent user experience.

The AI system supports flexible LLM integration via vLLM, allowing organizations to switch between various open-source models such as Llama, Gemma, ChatGPT, and Qwen. While the platform enables this flexibility, digiDations recommends using the same LLM as that adopted by the official TARA AI configuration to ensure alignment with tested prompt-response patterns.

For full feature support, users are also encouraged to adopt the AI system validated and maintained by digiDations. This ensures consistent capability delivery, timely updates, and integration with evolving validation use cases across threat intelligence, scenario design, and post-simulation insight generation.

4.5 AI Deployment Options and Supported Functionality

The ATLAS Security Validation Platform supports multiple LLM deployment options to accommodate different infrastructure, security, and compliance requirements.

Deployment Option 1: On-Premises Platform with digiDations-Hosted LLM
The ATLAS Console and its modules run entirely within the client's environment, on their own infrastructure (physical or virtual) or in their private cloud, while TARA AI queries are forwarded over a secure API to digiDations' managed LLM service. This approach delivers the full breadth of TARA functionality (scenario generation, rule enrichment, remediation guidance, threat analysis) without requiring any local AI infrastructure. Clients need only permit outbound HTTPS access to our API endpoint and maintain an active LLM license.

Deployment Option 2: SaaS with Integrated LLM
When clients subscribe to the ATLAS SaaS offering and hold a valid LLM license, all TARA AI features are available out of the box, such as scenario generation, rule enrichment, remediation guidance, and threat analysis. The LLM engine and its updates are fully managed by digiDations, ensuring clients always benefit from the latest model improvements and prompt optimizations.

Deployment Option 3: On Premises Platform with Customer's LLM
ATLAS and TARA AI run entirely within the clients' infrastructure, on physical servers, virtual machines, or in their private cloud, and connect to an LLM instance the client or a trusted third party manage. All TARA capabilities (scenario generation, rule enrichment, remediation guidance, threat analysis) are supported. However, output quality and feature coverage may vary based on the model's training and available capabilities.

4.6 AI-Powered Threat Intelligence Analysis

In addition to powering interactive features within the ATLAS platform, large language models (LLMs) are also integrated into digiDations' internal threat intelligence workflows. These models significantly enhance the speed, accuracy, and scalability of threat data processing across a variety of unstructured sources.

LLMs support the digiDations research team by parsing security reports, technical forums, malware analyses, and intelligence data from the dark web and social platforms. From these sources, the models automatically extract key information such as CVE identifiers, impacted assets, attacker groups, and relevant tactics, techniques, and procedures (TTPs). Compared to traditional rule-based pipelines, this approach increases the volume of actionable insights while reducing manual effort.

During Advanced Persistent Threat (APT) analysis, LLMs assist in correlating similar incidents, reconstructing attack chains, and inferring adversary intent based on observed activity. The models can map these findings directly to standardized frameworks like MITRE ATT&CK, helping analysts understand how threats align with known techniques and stages in the attack lifecycle.

For phishing analysis, LLMs can examine email content, attachments, and subject lines to identify social engineering patterns. These include urgency cues, impersonation techniques, and hidden malicious links, even in the absence of known indicators. LLMs can also simulate multi-turn conversations to assess the sophistication of phishing attempts and predict their impact.

When enhanced with Retrieval-Augmented Generation (RAG), the models incorporate contextual information from trusted knowledge sources, ensuring up-to-date and relevant output. This combination allows the system to remain responsive to emerging threats while grounding its responses in verified data.

By integrating these AI capabilities into threat intelligence workflows, digiDations improves the depth and timeliness of analysis while reducing operational burden. These enhancements directly support the continuous enrichment of the ATLAS validation library and reinforce the platform's ability to simulate and evaluate real-world threats.

Key Advantages of the ATLAS Platform

Modern IT environments are dynamic and complex, with changes in system configurations, application behavior, and network structure occurring frequently. These changes may go undetected by traditional security teams, creating gaps in visibility and undermining the effectiveness of existing defenses. To maintain an accurate and current understanding of the security posture, continuous monitoring and validation are essential.

The digiDations ATLAS Security Validation Platform is designed to detect and adapt to these changes in real time. It allows organizations to schedule attack simulations at regular intervals :weekly, daily, or even hourly—targeting specific behaviors or controls. When deviations in defense capabilities are detected, the platform automatically alerts the security operations team. This enables timely remediation and reinforces proactive defense.

Unlike traditional Breach and Attack Simulation (BAS) tools that focus narrowly on predefined techniques, the digiDations platform takes a comprehensive approach. It not only verifies the effectiveness of individual security controls but also assesses their interaction and cumulative impact on the broader security ecosystem. This helps establish a robust and continuously evolving defense baseline.

Key differentiators of the digiDations ATLAS platform include:

- **Deep Threat Expertise:** Over a decade of experience tracking APT groups, extracting malicious samples, and converting them into actionable simulation scenarios.
- **Advanced Threat Intelligence**: Real-time alignment with threat feeds and intelligence sources enables the platform to simulate current adversary TTPs and POCs as they emerge.
- **Safe Destructive Behavior Testing:** Unlike other solutions, digiDations offers controlled execution of high-risk behaviors without compromising production environments.
- **Drift Detection Capabilities:** The platform uniquely identifies security drift across environments, signaling changes that weaken or invalidate previously validated controls.
- **Operational Impact Analysis:** Through simulated attacks, the platform evaluates not just security posture, but also the downstream impact of vulnerabilities on business operations and resilience.

With weekly updates to its validation library and support for customizable scenarios, the platform ensures organizations stay aligned with evolving threats. More than a testing tool, the ATLAS platform enables a closed-loop validation process: identify gaps, validate

defenses, implement improvements, reverify effectiveness, and maintain a high-confidence baseline over time.

Data Protection Best Practices

The digiDations ATLAS Security Validation Platform enforces strict controls over access to sensitive data, including validation results, malicious payloads, and platform configurations. To further enhance data protection, organizations should implement the following operational safeguards:

- **Access Control**
  Limit access to sensitive data, simulation outcomes, and system configurations to authorized personnel only. Implement role-based permissions and enforce the principle of least privilege.
- **Third-Party Service Restrictions**
  Prohibit external vendors or unauthorized parties from performing platform maintenance or data access activities unless explicitly approved through a formal process.
- **Audit Logging and Review**
  Regularly audit platform access logs and validation activity to detect unusual behavior or unauthorized attempts to retrieve sensitive information.
- **Confidentiality Agreements**
  Require all internal teams and external collaborators with access to validation data or simulation content to sign confidentiality agreements that govern responsible use and disclosure.

These practices help safeguard operational integrity, reduce the risk of data leakage, and ensure compliance with internal policies and regulatory standards.

About digiDations

digiDations is an AI Native cybersecurity company helping organizations replace security assumptions with continuous, measurable validation. The question it answers is not whether an organization has vulnerabilities. It is whether their defenses can keep up with the threats targeting them right now.

Founded by cybersecurity veterans and AI researchers, digiDations built its product portfolio on Digital Mind, a proprietary AI foundation that continuously collects, correlates, and operationalizes real-world attack intelligence. The AI-Driven AEV portfolio includes ATLAS for Security Validation, APOLLO for Continuous Automated Red Teaming, and HELIOS for External Attack Surface Management. ORION for Threat Intelligence extends digiDations' CTEM coverage with adversary intelligence and visibility into emerging attack surfaces.

digiDations is building toward Autonomous Cyber Defense with TARA AI, the Autonomous CTEM Agent that self-directs across all four products: determining which scenarios to run, orchestrating execution, interpreting results, and generating reports, with teams approving decisions rather than defining the workflow.