Network Working Group   T. Madaras
Internet-Draft   GuardSuite, a VXMSecure Co.
Intended status: Experimental   P. Estis
Expires: June 2, 2026   GuardSuite, a VXMSecure Co.
December 2, 2025

GuardNet Authorization Protocol (GNA): Pre-Execution Authorization for High-Risk Digital Actions
draft-madaras-guardsuite-gna-00

Abstract

This document specifies the GuardNet Authorization Protocol (GNA), a pre-execution authorization framework for high-risk digital and physical actions. GNA defines an enforcement model in which sensitive operations (for example, payments, administrative changes, OT/ICS commands, and AI-initiated actions) are intercepted before execution, evaluated against policy and context, routed to one or more authenticated approvers, and executed only after a valid authorization token is issued.

GNA provides non-repudiable authorization receipts and is designed to integrate with existing identity providers and logging systems without requiring architectural changes. The protocol provides stronger authorization guarantees using digital signatures and contextual decisioning and is intended as a complement to, not a replacement for, existing authentication and monitoring systems.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 2, 2026.

Table of Contents

1. Introduction

Modern digital systems authenticate who a user is but rarely verify whether a specific high-risk action should be allowed at the moment of execution. Once a user or service is authenticated, many systems assume that all subsequent actions taken during that session are legitimate. This creates a gap between authentication and actual authorization at the action level.

The absence of a pre-execution authorization layer contributes to incidents such as privilege misuse, fraudulent financial transfers, misconfigurations of critical infrastructure, unsafe AI-driven actions, and other high-impact events. GNA introduces a standardized way to intercept and evaluate such actions before they execute, requiring explicit approval and producing a non-repudiable audit trail.

GNA is designed to be transport-agnostic and applicable across multiple domains, including financial systems, cloud infrastructure, operational technology (OT), and AI systems. It does not replace identity providers or logging systems; instead, it provides a complementary enforcement plane focused on high-risk actions.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119] and RFC 8174 [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

The following terms are used throughout this document:

RS (Requesting System)
: A system, application, service, or agent that initiates a high-risk action requiring pre-execution authorization.

AS (Authorization Service)
: The GuardNet Core service responsible for evaluating policy and context and issuing authorization decisions and tokens.

UA (User Agent)
: A secure device-based approver agent (for example, GuardID) used by human approvers to review and approve or deny an action.

Action
: A high-risk operation that MUST be authorized before execution, such as a large payment, critical configuration change, OT/ICS control operation, or AI-initiated task.

Challenge
: A request sent from the Authorization Service to one or more User Agents asking for an authorization decision on a specific Action.

Decision
: An APPROVE or DENY response returned from a User Agent to the Authorization Service, typically accompanied by a digital signature and relevant metadata.

Receipt
: A non-repudiable, tamper-evident record created by the Authorization Service that binds the Action, the applied policy, and the approver Decisions.

Execution Token
: A token issued by the Authorization Service that proves an Action was authorized and MAY now be executed by the Requesting System.

4. Architecture and Roles

GNA follows a model similar to Policy Enforcement Point (PEP) and Policy Decision Point (PDP) architectures common in access control systems, with components specialized for pre-execution action authorization:

Policy Enforcement Point (PEP)
: The GuardNet Edge Hook that intercepts Actions in the Requesting System and forwards authorization requests to the Authorization Service.

Policy Decision Point (PDP)
: The GuardCore Authorization Engine that evaluates policies, risk, and context, issues Challenges to User Agents, and returns an Execution Token upon approval.

Approver Trust Anchor
: The User Agent (for example, GuardID) which provides a trusted interface for human approvers and signs Decisions.

A typical end-to-end flow is:

1. The Requesting System invokes the PEP when a high-risk Action is requested.

2. The PEP constructs an AUTH-REQ message and sends it to the PDP (Authorization Service).

3. The PDP evaluates policies and context and, if required, sends AUTH-CHALLENGE messages to one or more User Agents.

4. User Agents present the Action context to human approvers and return AUTH-DECISION messages (APPROVE or DENY).

5. The PDP aggregates Decisions, creates a Receipt, and issues an AUTH-RESULT containing an Execution Token if the Action is approved.

6. The PEP validates the Execution Token and either allows or blocks execution of the Action in the Requesting System.

5. Protocol Overview

GNA defines a small set of logical messages:

- AUTH-REQ: Sent by the PEP to the PDP to request authorization for a specific Action.

- AUTH-CHALLENGE: Sent by the PDP to one or more UAs, requesting a Decision.

- AUTH-DECISION: Sent by a UA to the PDP to convey an APPROVE or DENY decision.

- AUTH-RESULT: Sent by the PDP to the PEP, containing an Execution Token and a result (approved or denied).

- RECEIPT: Created by the PDP and persisted to a logging or ledger system for non-repudiation and audit.

GNA does not mandate specific transport protocols. Implementers MAY choose HTTPS/REST, gRPC, message queues, or other secure transports

as appropriate, provided that confidentiality, integrity, and authenticity are preserved.

6. Messages and Data Structures

This section describes the abstract fields associated with the various messages. Concrete encodings (for example, JSON, CBOR, protobuf) are left to profiles and future documents.

6.1. AUTH-REQ

AUTH-REQ is constructed by the PEP and sent to the PDP when a high-risk Action is initiated. It SHOULD include:

- action_id: Unique identifier for the Action.
- action_type: Category or type of Action (for example, "payment", "config_change", "ot_command").
- requester_identity: Identity of the initiator (derived from an existing identity provider).
- context: Metadata such as amount, target resource, environment, and location.
- client_metadata: Optional device, application, or session attributes.

6.2. AUTH-CHALLENGE

AUTH-CHALLENGE is sent from the PDP to one or more UAs when policy requires human or multi-party approval. It SHOULD include:

- challenge_id: Unique identifier for this Challenge.
- action_summary: Human-readable description of the Action.
- policy_id: Identifier of the policy being applied.
- context_snapshot: Relevant context at the time of the request.
- nonce: Random value to prevent replay.

6.3. AUTH-DECISION

AUTH-DECISION is sent from a UA to the PDP to convey an approval or denial. It SHOULD include:

- challenge_id
- decision: APPROVE or DENY.

- approver_identity
- signature: Digital signature over the decision and key context.
- timestamp

6.4. AUTH-RESULT

AUTH-RESULT is sent from the PDP back to the PEP. It MUST indicate whether the Action is authorized and SHOULD carry an Execution Token if approved. It SHOULD include:

- action_id
- result: APPROVED or DENIED.
- execution_token: Token the PEP can validate prior to executing the Action.
- expiry: Time after which the token is no longer valid.

6.5. RECEIPT

A Receipt is generated by the PDP after processing the Action and any required approvals. It is intended for logging and audit and is not normally transmitted inline to RS or UA. It SHOULD include:

- receipt_id
- action_hash: Hash of the Action parameters.
- policy_hash: Hash of the policy used.
- approver_identities
- decision_summary
- timestamp
- signature: Digital signature over the Receipt.

7. Policy and Materiality Evaluation

GNA assumes a configurable policy layer that governs which Actions require authorization and what form that authorization MUST take. Policies MAY define:

- Thresholds (for example, amounts or scope of change).
- Required approver roles or identities.
- N-of-M approval schemes.

- Segregation-of-duties requirements.
- Contextual conditions such as time of day, location, or device posture.

The exact policy language and storage is out of scope for this document, but GNA is designed to work with existing PDP tools as well as custom policy engines.

8. Security Considerations

GNA is designed to reduce risk arising from:

- Compromised credentials used to execute high-value actions.
- Malicious insiders acting alone with excessive privileges.
- Deepfake and social-engineering attacks that coerce execution of harmful actions.
- Automated or AI-driven systems operating without human oversight.

Implementers MUST consider:

- Protection of private keys for signing Decisions and Receipts.
- Hardening of PEPs and PDP against bypass or tampering.
- Availability strategies for the PDP (for example, fail-open versus fail-closed behavior for different classes of Actions).

GNA does not prevent collusion between multiple approvers or compromise of both a Requesting System and its corresponding approver devices. Those risks MUST be addressed with broader organizational controls and endpoint security.

9. IANA Considerations

This document makes no requests of IANA.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Authors' Addresses

Tom Madaras
GuardSuite, a VXMSecure Company
Hollywood, FL 33021
USA

Email: tom@vxmsecure.com

Pat Estis
GuardSuite, a VXMSecure Company
Houston, TX
USA

Email: pat.estis@gmail.com