saag
Internet-Draft
Intended status: Informational
Expires: 22 April 2026

B. Chen
OpenNHP

19 October 2025

Network infrastructure Hiding Protocol
draft-opennhp-ace-nhp-01

Abstract

The Network infrastructure Hiding Protocol (NHP) is a cryptography-based session-layer protocol designed to implement Zero Trust principles by rendering protected network resources invisible to unauthorized entities. By requiring authentication before connection and operating at OSI layers 5 , NHP conceals IP addresses, ports, and domains from exposure to reconnaissance and automated exploitation, effectively reducing the attack surface. This draft defines the architecture, message format, and workflow of the NHP protocol, outlines its security objectives, and provides guidance for integration into modern network infrastructures and Zero Trust deployments.

title: "Network-Infrastructure Hiding Protocol (NHP)" abbrev: "NHP" docname: draft-opennhp-ace-nhp-01 category: informational stream: independent submissiontype: independent number: 00 date: 2025-10-19 v: 1 area: "Security" workgroup: "secdp" keyword:

* zero trust
* session layer
* network obfuscation venue: group: "saag" type: "Independent Submission" mail: "saag@ietf.org (mailto:saag@ietf.org)" arch: "https://mailarchive.ietf.org/arch/browse/secdp/ (https://mailarchive.ietf.org/arch/browse/secdp/)" github: "OpenNHP/ietf-rfc-nhp" latest: "https://OpenNHP.github.io/ietf-rfc-nhp/draft-opennhp-ace-nhp.html (https://OpenNHP.github.io/ietf-rfc-nhp/draft-opennhp-ace-nhp.html)"

author:

* fullname: Benfeng Chen organization: OpenNHP email: benfeng@gmail.com (mailto:benfeng@gmail.com)

normative:

* RFC2119

* RFC8174
* RFC9000
* RFC8446
* RFC9180
* NoiseFramework

informative:

* NIST.SP.800-207
* CSA.SDP.Spec2.0
* CSA.SDP.Architecture.3.0
* CSA.ZeroTrust.GuidingPrinciples
* CSA.AsymmetricCrypto.ZT
* OpenNHP.Readme

The Network-Infrastructure Hiding Protocol (NHP) is a cryptography-based session-layer protocol designed to operationalize Zero Trust principles by concealing protected network resources from unauthorized entities. NHP enforces authentication-before-connect access control, rendering IP addresses, ports, and domain names invisible to unauthorized users. This document defines the protocol architecture, cryptographic framework, message format, and workflow to enable independent implementation of NHP. It also provides guidance for integration with Software-Defined Perimeter (SDP), DNS, and Zero Trust policy engines.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://OpenNHP.github.io/ietf-rfc-nhp/draft-opennhp-ace-nhp.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-opennhp-ace-nhp/.

Discussion of this document takes place on the WG Working Group mailing list (mailto:saag@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ace/. Subscribe at https://www.ietf.org/mailman/listinfo/saag/.

Source for this draft and an issue tracker can be found at https://github.com/OpenNHP/ietf-rfc-nhp.

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 22 April 2026.

Copyright Notice

Table of Contents

1. 1. Introduction

Since its inception in the 1970s, the TCP/IP networking model has prioritized openness and interoperability, laying the foundation for the modern Internet. However, this design philosophy also exposes systems to reconnaissance and attack.

Today, the cyber threat landscape has been dramatically reshaped by the rise of AI-driven attacks, which bring unprecedented speed and scale to vulnerability discovery and exploitation. Automated tools continuously scan the global network space, identifying weaknesses in real-time. As a result, the Internet is evolving into a "Dark Forest," where *visibility equates to vulnerability*. In such an environment, any exposed service becomes an immediate target.

The Zero Trust model, which mandates continuous verification and eliminates implicit trust, has emerged as a modern approach to cybersecurity. Within this context, the Network infrastructure Hiding Protocol (NHP) offers a new architectural element: authenticated-before-connect access at the session layer. Inspired by Cloud Security Alliance's Single-Packet Authorization (SPA) and Software Defined Perimeter (SDP) technologies, NHP advances the concept further by using cryptographic protocols (e.g., Noise, ECC) to obfuscate infrastructure and enforce granular access control.

This document outlines the motivations behind NHP, its design objectives, message structures, integration options, and security considerations for adoption within Zero Trust frameworks.

2. 2. Conventions and Definitions

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 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

NHP: Network-Infrastructure Hiding Protocol SPA: Single-Packet Authorization SDP: Software-Defined Perimeter ZTA: Zero Trust Architecture ECC: Elliptic Curve Cryptography AEAD: Authenticated Encryption with Associated Data NAC: Network Access Control

3. 3. Design Objectives

The NHP protocol is designed to:

* Eliminate unauthorized network visibility by enforcing authentication prior to session establishment.

* Operate at the OSI Session Layer, complementing existing TCP, UDP, and QUIC transports.

* Support decentralized trust using asymmetric cryptography and ephemeral key exchange.

* Enable fine-grained, context-based policy enforcement across heterogeneous environments.

* Integrate with existing Zero Trust controllers, SDP gateways, and identity systems.

4. 1. Introduction

Since the 1970s, TCP/IP has provided universal connectivity but also exposed every reachable service to reconnaissance and attack. Modern AI-driven reconnaissance and exploitation tools can automatically identify and compromise exposed endpoints, turning Internet visibility into vulnerability. This document introduces the Network-Infrastructure Hiding Protocol (NHP), a session-layer mechanism that enforces authenticated-before-connect access, ensuring that only authorized clients can detect and reach protected resources.

NHP builds upon foundational work in the Cloud Security Alliance' s Software-Defined Perimeter (SDP) and Single-Packet Authorization (SPA) frameworks, extending them with modern asymmetric cryptography and Noise Protocol-based mutual authentication. NHP replaces traditional perimeter visibility with a cryptographically controlled, context-aware trust fabric.

5. 2. Terminology and Conventions

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 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

NHP: Network-Infrastructure Hiding Protocol SPA: Single-Packet Authorization SDP: Software-Defined Perimeter ZTA: Zero Trust Architecture ECC: Elliptic Curve Cryptography AEAD: Authenticated Encryption with Associated Data NAC: Network Access Control

6. 3. Design Objectives

The NHP protocol is designed to:

* Eliminate unauthorized network visibility by enforcing authentication prior to session establishment.

* Operate at the OSI Session Layer, complementing existing TCP, UDP, and QUIC transports.

* Support decentralized trust using asymmetric cryptography and ephemeral key exchange.

* Enable fine-grained, context-based policy enforcement across heterogeneous environments.

* Integrate with existing Zero Trust controllers, SDP gateways, and identity systems.

7. 4. Architectural Overview

NHP operates as a distributed session-layer protocol that enforces authentication-before-connect access between clients and protected resources. The architecture includes three primary components:

* *NHP-Agent* a client-side process or SDK that initiates communication with the protected network by generating and sending an NHP-KNK message to the NHP-Server. It performs cryptographic key exchange and authentication using Noise-based handshakes.

* *NHP-Server* the core control-plane service that receives and validates NHP-KNK messages, authenticates the NHP-Agent, and determines access decisions. The NHP-Server interfaces with external *Authorization Service Providers (ASP)* or IAM systems to evaluate identity, context, and policy. It also manages

communication with NHP-AC components, instructing them to open or close access paths. Functionally, the NHP-Server maps to the *Policy Administrator* role defined in *NIST SP 800-207 Zero Trust Architecture*, responsible for policy evaluation and decision-making.

* *NHP-AC (Access Controller)* the enforcement component residing logically or physically near protected resources. Upon receiving an NHP-AOP command from the NHP-Server, the NHP-AC updates its access control tables, temporarily allowing the NHP-Agent to reach the protected resource. It automatically reverts to the default-deny state when the session expires. The NHP-AC corresponds to the *Policy Enforcement Point (PEP)* in NIST SP 800-207 terminology.

In this model, authentication, authorization, and access enforcement are strictly decoupled. The NHP-Agent never directly accesses the NHP-AC without prior validation by the NHP-Server. This structure supports horizontal scalability and allows multiple NHP-Servers and NHP-ACs to operate in cluster configurations for redundancy and high availability.

7.1. 4.1 Component Interactions and Deployment Models

NHP components can be deployed in different configurations depending on the size and topology of the protected network:

* *Standalone Deployment:* For small environments or testing scenarios, the NHP-Server and NHP-AC can coexist on the same host. This configuration simplifies setup while maintaining full protocol compliance.

* *Clustered Deployment:* In enterprise or cloud environments, multiple NHP-Servers can be deployed in a load-balanced cluster. Each server manages a pool of NHP-AC instances distributed across data centers or network segments. The NHP-Agent dynamically discovers the nearest NHP-Server through DNS or bootstrap configuration.

* *Edge AC Deployment:* Edge nodes (e.g., gateways, routers, or micro-segmentation agents) can host lightweight NHP-AC components. These edge ACs enforce fine-grained policies close to workloads, improving latency and fault isolation.

* *Multi-Tenant Deployment:* In service-provider or multi-cloud environments, each tenant can operate an independent NHP-Server while sharing an underlying AC infrastructure. The NHP protocol's namespace isolation ensures complete tenant separation through identity-scoped keys and per-tenant policy databases.

This modular architecture provides flexibility for diverse Zero Trust environments—from on-premises datacenters to global cloud infrastructures—while maintaining the core NHP principles of invisibility, least privilege, and continuous verification.

8. 5. Protocol Workflow

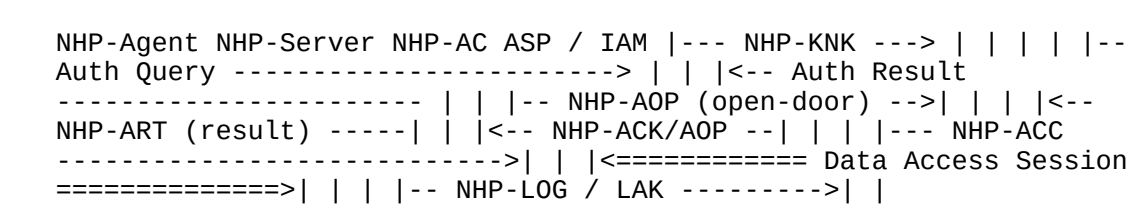
8.1. 5.1 Control Plane vs Data Plane

The *Control Plane* carries cryptographic authentication and authorization information among NHP-Agent, NHP-Server, NHP-AC, and optional external *Authorization Service Providers (ASP)*. The *Data Plane* carries application data between the resource requester (NHP-Agent host) and the protected resource, but only after NHP-AC explicitly authorizes access. This strict separation enforces the _authenticate-before-connect_ principle.

8.2. 5.2 NHP Workflow Steps

1. NHP-Agent sends *NHP-KNK* to *NHP-Server*.
2. NHP-Server validates and queries *ASP*.
3. ASP returns authorization decision.
4. NHP-Server sends *NHP-AOP* to *NHP-AC*.
5. NHP-AC enforces and replies with *NHP-ART*.
6. NHP-Server sends *NHP-ACK/AOP* to *NHP-Agent*.
7. NHP-Agent connects using *NHP-ACC*.
8. NHP-Server and NHP-AC maintain *Keepalive/Logs*.
9. Logs uploaded for compliance and auditing.

8.3. 5.3 Sequence Diagram



9. 6. Cryptographic Framework

NHP employs the *Noise Protocol Framework* using the *XX* handshake pattern by default for full forward secrecy and identity protection, or *K* pattern for performance-optimized one-way initiation. Recommended primitives:

* DH function: Curve25519
* Cipher: ChaCha20-Poly1305
* Hash: SHA-256
* Key Derivation: HKDF

10. 7. Message Structure

Field	Size	Description
Version	1 byte	Protocol version (0x01)
Type	1 byte	Message type code
Flags	1 byte	Control flags
Nonce	12 bytes	AEAD nonce for replay protection
Timestamp	8 bytes	UNIX epoch time (ms)
Payload Length	2 bytes	Encrypted payload length

Table 1

11. 8. Security Considerations

NHP ensures infrastructure invisibility by encrypting all handshake traffic and performing mutual authentication before connection establishment. Implementations MUST use strong ECC keys, prevent replay, and maintain session expiration enforcement.

12. 9. IANA Considerations

This document has no IANA actions.

13. 10. Reference Implementation

An open-source reference implementation of NHP is available at: *https://github.com/OpenNHP/opennhp (https://github.com/OpenNHP/opennhp)*

14. 11. Acknowledgments

This work extends concepts from the Cloud Security Alliance SDP WG and integrates guidance from NIST SP 800-207 and CSA Zero Trust research.

15. 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/rfc/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/rfc/rfc8174>.

Appendix A. Normative References

[RFC2119] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, 1997. [RFC8174] Leiba, B., “Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words”, 2017. [NoiseFramework] Perrin, T., “The Noise Protocol Framework”, 2018.

Appendix B. Informative References

[NIST.SP.800-207] Rose, S., Borchert, O., Mitchell, S., and Connelly, S., “Zero Trust Architecture”, 2020. [CSA.SDP.Spec2.0] CSA, “Software Defined Perimeter Specification v1.0”, 2021. [CSA.SDP.Architecture.3.0] CSA, “SDP Architecture Guide v3.0”, 2025. [CSA.ZeroTrust.GuidingPrinciples] CSA, “Zero Trust Guiding Principles”, 2024. [CSA.AsymmetricCrypto.ZT] CSA, “Using Asymmetric Cryptography to Help Achieve Zero Trust Objectives”, 2024. [OpenNHP.Readme] Chen, B., “OpenNHP: Open Source Zero Trust Security Toolkit”, GitHub, 2025.

Appendix C. IANA Considerations

This document has no IANA actions.

Acknowledgments

This work builds upon foundational research from the Cloud Security Alliance (CSA) (https://cloudsecurityalliance.org/) and benefits from the collaborative support of the China Computer Federation (CCF) (https://www.ccf.org.cn/en/). The authors would also like to thank the OpenNHP (https://github.com/OpenNHP/opennhp) open source community for their contributions, testing, and feedback on early implementations of the Network infrastructure Hiding Protocol (NHP).

Author’s Address

Benfeng Chen
OpenNHP
Email: benfeng@gmail.com