N

NDR: Network Detection and Response Explained

Learn how NDR analyzes network traffic and metadata, where sensors get visibility, what encrypted traffic reveals, and how NDR differs from IDS, EDR, and XDR.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
3 min

CONTENTS

In this guide

01 What does NDR stand for?
02 Data sources used by NDR
03 What NDR can detect
04 Encrypted traffic visibility
05 NDR versus IDS, IPS, and firewall
06 NDR versus EDR and XDR
07 Deployment pitfalls
08 Deployment and response checklist

Network Detection and Response (NDR) collects and analyzes network traffic or network-event data to identify suspicious behavior, support investigations, and guide containment. NDR can reveal communication between systems even when an endpoint

sensor is missing, but its visibility depends on where traffic is observed and what metadata or content is available.

01 What does NDR stand for?

NDR stands for **Network Detection and Response**. It observes communications rather than relying only on an agent installed on each endpoint. The “response” portion may alert an analyst, enrich an incident, request a firewall or access-control change, or integrate with another platform; it does not mean every NDR product blocks traffic inline.

02 Data sources used by NDR

- Packets from a network tap, switch mirror/SPAN port, virtual switch, or cloud traffic mirror.
- Flow records such as NetFlow, IPFIX, VPC flow logs, or equivalent cloud telemetry.
- DNS, DHCP, proxy, firewall, VPN, and identity context.
- TLS, protocol, timing, volume, and connection metadata.
- Asset inventory and threat intelligence used to enrich observations.

03 What NDR can detect

- Unexpected internal scanning and lateral movement.
- Beaconing or periodic command-and-control connections.
- Unusual data-transfer volume, destination, protocol, or time.
- Connections to known malicious infrastructure.
- Protocol misuse, tunneling, or communication inconsistent with an asset's role.
- Activity from unmanaged, embedded, operational-technology, or guest devices.

An anomaly is not automatically malicious. Backups, vulnerability scanners, software distribution, monitoring, and business batch jobs can resemble attack behavior.

04 Encrypted traffic visibility

NDR often cannot read the application payload of properly encrypted traffic. It may still analyze source and destination, certificate details, protocol fingerprints, DNS context, packet sizes, timing, and session behavior. TLS interception can provide more content visibility but introduces privacy, certificate-management, legal, performance, and security risks. It should be limited to an approved architecture and policy.

05 NDR versus IDS, IPS, and firewall

Control	Primary role
IDS	Detects suspicious network activity and generates alerts, often from signatures and protocol rules.
IPS	Operates inline and can block selected traffic automatically.
Firewall	Enforces connection policy using addresses, ports, state, identity, or application information.
NDR	Combines network telemetry, analytics, investigation context, hunting, and response workflows.

06 NDR versus EDR and XDR

- **EDR** provides deep process, file, user, and response visibility on onboarded endpoints.
- **NDR** observes communication across selected network paths and can cover devices without agents.
- **XDR** correlates multiple security domains and may consume both EDR and NDR signals.

These controls are complementary. A network alert gains meaning when it can be tied to the responsible process, account, email, or cloud event.

07 Deployment pitfalls

- A mirror port is oversubscribed and silently drops packets.
- East-west, cloud, remote-user, or encrypted-tunnel traffic bypasses sensors.
- NAT or missing DHCP/identity context prevents reliable asset attribution.
- High-volume storage and retention are planned without a clear investigation need.
- Baseline learning treats an existing compromise as normal.

08 Deployment and response checklist

1. Map critical traffic paths and identify visibility gaps before placing sensors.
2. Synchronize time and enrich network data with asset, identity, DNS, and cloud context.
3. Validate packet loss, flow completeness, and retention.
4. Baseline known scanners, backups, proxies, update services, and business workflows.
5. Define who can capture content and how sensitive traffic is protected.
6. Test response actions such as firewall blocks, endpoint isolation, and credential revocation with human approval appropriate to risk.

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com