How AI Driven NDR Platforms Provide Evidence Through Packets

qa|cafe

A Strategic Guide for Product Managers to Turn Packet Analysis into a Competitive Advantage

SECURITY PLATFORMS HAVE NEVER BEEN MORE INTELLIGENT

Modern NDR and traffic analytics solutions detect lateral movement, identify encrypted threats, surface anomalies, and correlate behavior at scale. Machine learning models grow more refined each year. Detection quality improves release after release.

And yet, during evaluations, renewals, and incident escalations, security buyers still ask a deceptively simple question:

"Can you show me exactly what happened?"

That question illustrates a competitive opportunity: solutions that can say "yes" differentiate themselves clearly as a complete investigation platform.

But adding this to your product doesn't require your teams to divert resources or become packet-capture analysis experts. Packet Viewer is a set of embeddable React components backed by a Docker-hosted analysis engine—drop it into your existing application and your users get a full Wireshark-grade packet inspection experience without leaving your platform. You gain the ability to give your customers this view immediately and show them the evidence they want in the context of the information you provide.

This guide explains why leading security platforms are embedding packet-level evidence directly into their products, and how doing so easily with Packet Viewer becomes a durable competitive advantage.

The strategic shift from AI analytics to a complete solution

If you are building a modern NDR or security analytics platform, you already believe the packets matter. You ingest them, you parse them, you extract metadata from them. Your AI derives behavioral detections from them.

Security analytics generates insight. But insight is not the final deliverable. Enterprise buyers, MSSPs, and regulated organizations increasingly demand evidence:

- **Why was this flagged?**
- **What evidence supports this conclusion?**
- **Can we verify this independently?**

AI answers what your system believes is happening.
Packet evidence answers what actually happened.

But in many platforms, packet capture powers the engine but sits at the edge of the user workflow. It exists, but it behaves like a backend artifact rather than a key part of a first-class investigative tool.

WHERE MANY PLATFORMS STALL

Security platforms compete on the promise of AI powered detection and response. But without evidence of how it came to a conclusion, the promise feels unfinished. An alert or AI response without embedded evidence is helpful, but incomplete. An alert or AI response with in-context packet proof is operationally decisive.

Here is the common workflow in many security products:

1) The platform surfaces a suspicious behavior.

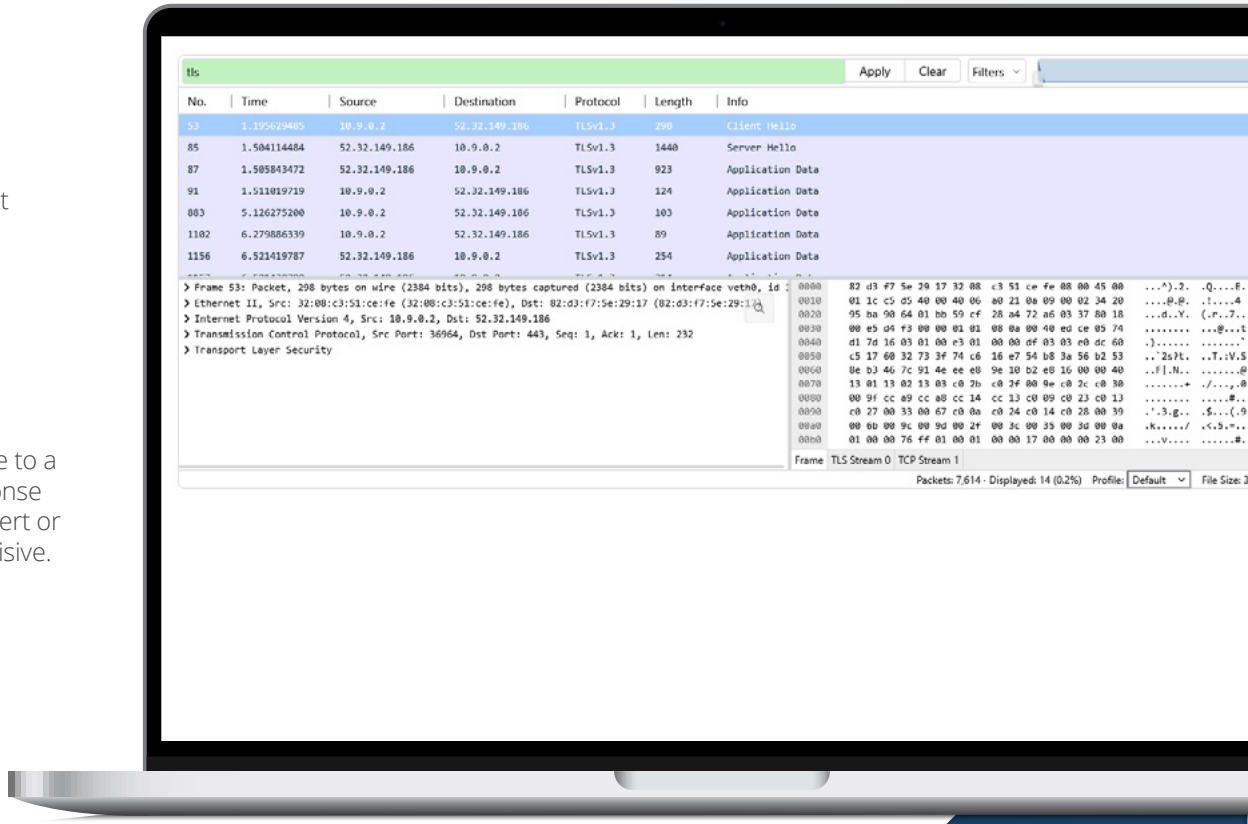
2) The analyst pivots to investigate.

3) The product offers a PCAP export.

4) The analyst downloads and opens Wireshark.

5) Context must be reconstructed manually.

The moment an analyst leaves your product to validate an AI-driven detection, trust shifts away from your platform and toward the external tool.



4 Hidden Costs of Fragmented Investigations

1. Losing Context
2. Product Narrative Break
3. Data Boundary Concerns
4. Investigation Inefficiency

1. Losing Context

The analyst transitions from a contextual, correlated environment into a raw packet view detached from the surrounding investigation. Time filters, user identity context, detection metadata, and correlated events must be mentally reconstructed. This increases the analyst's workload and reduces efficiency.

2. Product Narrative Break

From a buyer's perspective, the product implicitly says:

"WE CAN DETECT IT. BUT TO TRULY UNDERSTAND IT, USE SOMETHING ELSE."

That subtle handoff weakens perceived completeness. In competitive evaluations, this matters. Platforms that keep evidence in-context feel more cohesive and mature.

3. Data Boundary Concerns

Downloading PCAPs moves sensitive packet data onto analyst endpoints. For enterprise and regulated customers, this introduces:

- **Governance questions**
- **Storage concerns**
- **Audit complexity**
- **Increased exposure surface**

Even if risk is manageable, perception of control matters.

4. Investigation Inefficiency

When Tier 1 escalates to Tier 3, the workflow often fragments further:

- **Multiple PCAPs circulate**
- **Notes travel separately**
- **Context decouples from evidence**

The platform becomes just another tool in the chain, rather than an investigative center for the entire DFIR process.

In markets where detection quality increasingly converges, the platform that owns the entire investigative lifecycle (alert to evidence) has a distinct advantage.

Changing competitive positioning with in-app evidence

EMBEDDING PACKET ANALYSIS DIRECTLY INTO A SECURITY PLATFORM DOES MORE THAN ADD A FEATURE. IT RESHAPES HOW BUYERS PERCEIVE THE PRODUCT.

1. **It strengthens analyst confidence**

   Alert-to-evidence becomes a single motion. Analysts validate detections inside the same environment that generated them.

2. **It reduces false-positive disputes**

   Enterprise customers frequently challenge detections. Embedded packet evidence allows vendors to demonstrate findings immediately rather than describing them abstractly.

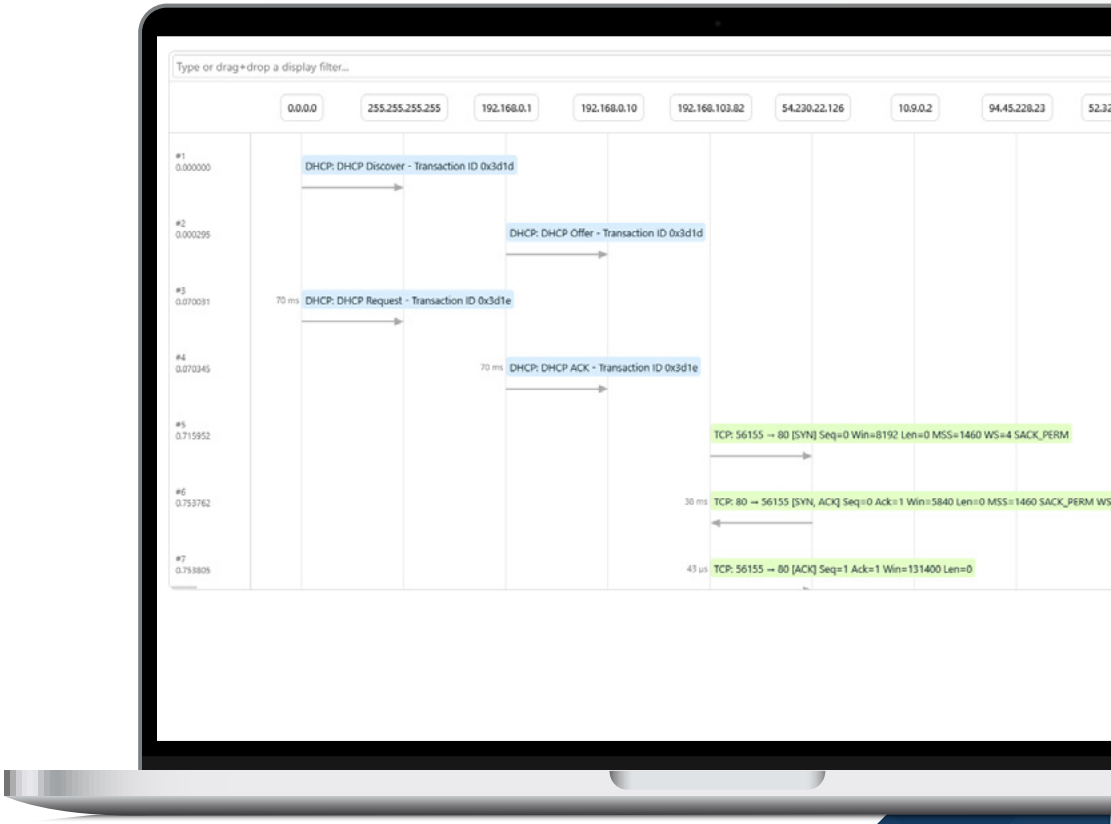
3. **It shortens incident response cycles**

   Every tool switch adds time. Keeping packet inspection in-context compresses investigation loops.

4. **It improves audit and escalation readiness**

   When incidents escalate to legal, compliance, or executive review, packet-level proof inside the platform strengthens credibility.

   In markets where dashboards and analytics increasingly converge, workflow quality becomes a source of differentiation. This competitive shift elevates packet evidence from a backend artifact to a first-class investigative experience. Your leading platform no longer treats PCAP export as the final step. It treats packet-level evidence as a natural part of the complete solution. And that must happen inside your product.



The strategic decision: build or buy?

Once product teams recognize the need for packet-level proof, they face a familiar fork in the road:

- **Build a browser-based packet analysis capability internally**
- **Rely on PCAP exports**
- **Run a remote desktop version of Wireshark**
- **Or embed a purpose-built solution**

The decision appears tactical. It is not. It shapes roadmap focus, engineering risk, and competitive timing.

Below is a strategic lens for evaluating the choice.

EXTRAHOP

A real-world example: how ExtraHop shows proof in the packets

In the NDR space, ExtraHop built its platform on full-packet visibility and advanced behavioral analytics. Its RevealX platform already analyzes traffic deeply to power high-quality detections.

But detection alone did not complete the investigative story.

Security teams still needed to:

- **Inspect raw packet evidence behind alerts**
- **Validate suspicious sessions and flows**
- **Confirm what data was transmitted**
- **Present defensible findings to stakeholders**

By embedding Packet Viewer directly into the RevealX experience, ExtraHop enabled packet-level inspection within the same investigative workflow.

The impact was strategic, not cosmetic:

- **Analysts stayed inside context**
- **Escalations moved faster**
- **Evidence became part of the product narrative**

YOU CAN READ THE FULL CASE STUDY HERE:
https://www.qacafe.com/resources/how-extrahop-uses-packet-viewer-to-bring-forensic-evidene-to-the-forefront/

The lesson is clear: leading NDR platforms treat packet evidence as an integrated capability—not an export artifact.

6 Strategic Risks of Building Packet Analysis In-House

1. Strategic Focus
2. Time to Market
3. Engineering Risk
4. Workflow Quality
5. Data Governance
6. Long-Term Maintenance

1. Strategic Focus

Packet analysis is rarely the core differentiator for NDR platforms. Detection quality, analytics depth, response automation, and ecosystem integration are.

Building a mature packet UI diverts engineering attention toward protocol tooling rather than analytics innovation.

Embedding preserves focus on the areas where you truly compete.

STRATEGIC QUESTION:

Is packet UI engineering where we want to spend our resources?

2. Time to Market

A Wireshark-grade browser experience requires:

- **Thousands of protocol parsers**
- **Filter engines**
- **Stream reassembly**
- **Linked packet grids, trees, and hexdumps**
- **Performance optimization**
- **Continuous protocol updates**

This is a multi-quarter initiative before reaching production maturity.

Embedding compresses that timeline dramatically.

STRATEGIC QUESTION:

Can we afford to delay competitive parity for a year?

3. Engineering Risk

Packet analysis is deceptively complex. Correctness matters. Analysts trust packet evidence more than dashboards.

An incomplete or partially mature implementation risks credibility.

Embedding leverages a mature, Wireshark-native backend with broad protocol support.

STRATEGIC QUESTION:

Do we want to own protocol correctness and evolution long term?

4. Workflow Quality

Many internal builds begin as “minimum viable” solutions. They often support only basic filtering or limited protocol inspection.

Advanced users still revert to Wireshark.

Embedding delivers a familiar, analyst-grade experience from day one.

STRATEGIC QUESTION:

Will power users trust our implementation enough to abandon external tools?

5. Data Governance

PCAP downloads move sensitive packet data onto analyst machines, increasing exposure surface and compliance questions.

Keeping packet evidence in-platform maintains boundary integrity and reduces governance complexity.

STRATEGIC QUESTION:

Does our current workflow introduce avoidable data exposure?

6. Long-Term Maintenance

Packet tooling is never finished.

Protocols evolve. Expectations rise. Performance demands increase.

Building means committing to permanent maintenance.

Embedding converts ongoing protocol evolution into predictable infrastructure rather than internal burden.

STRATEGIC QUESTION:

Are we prepared to maintain a packet analysis product indefinitely?

Using Packet Viewer as a competitive advantage

If the strategic shift in security markets is from dashboards and tools to AI analytics and response backed up by packet evidence, then the practical challenge becomes clear: how do you elevate packet-level evidence inside your platform without turning packet tooling into a parallel product effort?

Packet Viewer does this by embedding a Wireshark-native packet analysis experience directly into your application. Instead of investigations ending with a download, they conclude where they should: inside the same interface that generated the detection. Analysts can pivot from an AI output into full packet inspection in context. They can follow TCP streams, inspect TLS handshakes, analyze DNS queries, and review payloads without manually reconstructing the surrounding investigation.

Packet Viewer gives you powerful options with little to no development time on your side.

3 ways to deliver packet analysis without the complexity

1. **Ship a production-ready packet interface**
2. **Pivot seamlessly from alerts to packets**
3. **Keep packet data inside trusted boundaries**

1. Ship a production-ready packet interface

Packet Viewer provides a production-ready, analyst-grade packet interface that vendors can embed directly into their products. The interface is familiar to anyone who has used Wireshark, with the same core inspection model and deep protocol support. Analysts can immediately filter traffic, follow streams, inspect payloads, and analyze sessions without learning a new tool or adjusting their workflow. This means:

- **Immediate credibility with your customer's analysts**
- **No packet expertise required**
- **Rapid integration into your platform**
- **Reduced engineering burden**

From a product team's perspective, the benefit is simplicity. Packet Viewer delivers a complete packet analysis UI that drops into your platform as embeddable components backed by a lightweight container service. Your engineers do not need to implement protocol parsing, build filtering engines, or understand the nuances of packet reconstruction to deliver a credible investigative experience.

2. Pivot directly from AI output to verifiable packet proof

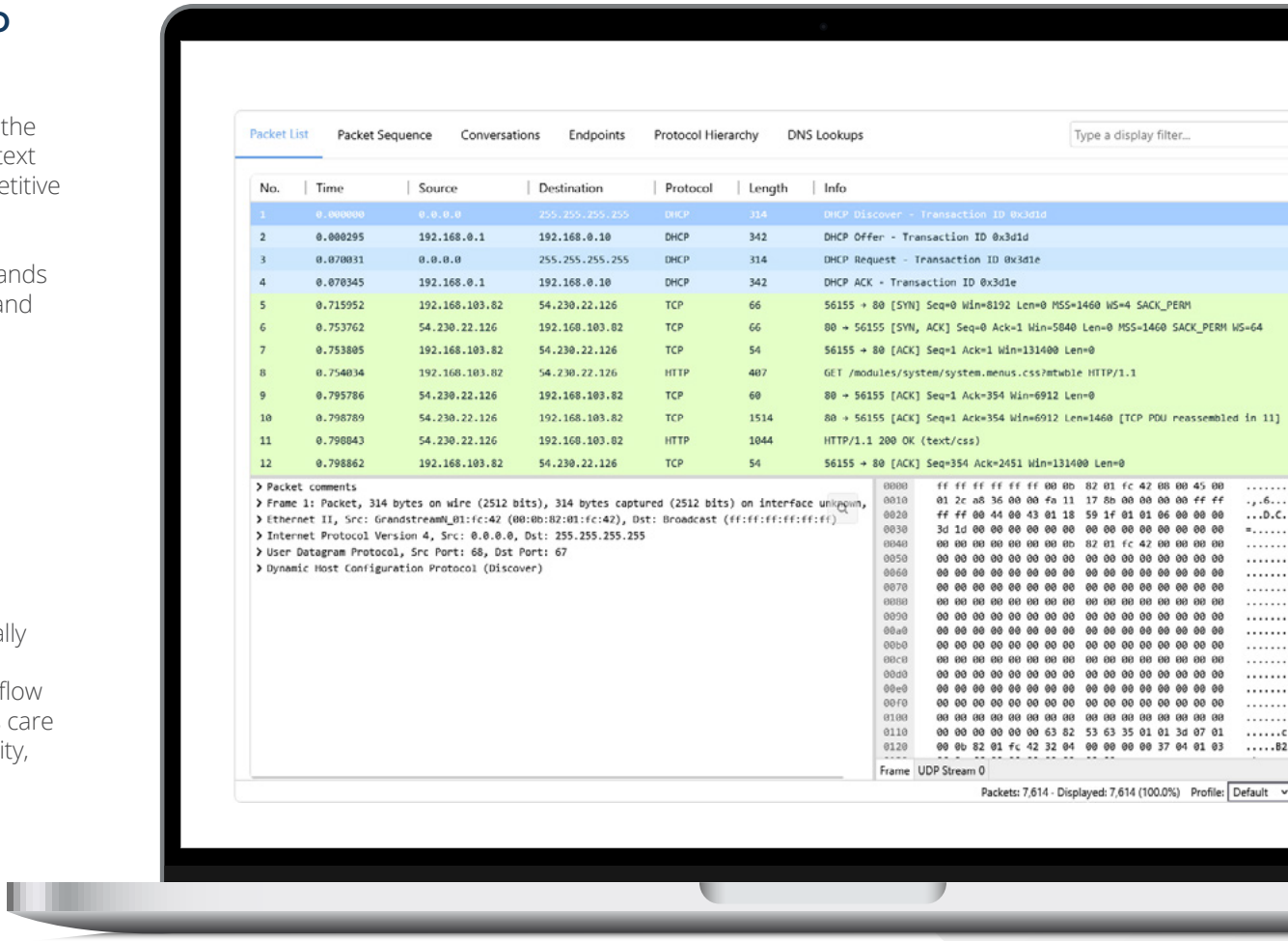
Packet Viewer lets you design workflows that use the information your solution already has to add context to mountains of packet data. This powerful competitive shift happens at a single interaction: the pivot.

When an analyst clicks an alert and immediately lands on the relevant packet evidence, already filtered and scoped, the product feels complete.

THAT PIVOT:

- **Preserves time context**
- **Preserves correlation context**
- **Preserves user focus**
- **Eliminates tool switching**

Instead of reconstructing the investigation manually after downloading a PCAP, analysts move directly from suspicion to validation. Over time, that workflow efficiency becomes evident in metrics that buyers care about: mean time to resolution, analyst productivity, and reduced escalation.



3. Keep packet data inside trusted boundaries

Security buyers scrutinize data movement as carefully as detection quality. Relying on “download PCAP” workflows introduces unnecessary exposure:

- **Sensitive traffic moves to analyst endpoints**
- **Copies of packet data proliferate**
- **Governance complexity increases**

Packet Viewer operates inside your infrastructure. The packet data remains where you want it. Analysts inspect traffic through your platform without exporting it to external SaaS services or downloading it locally unless absolutely required.

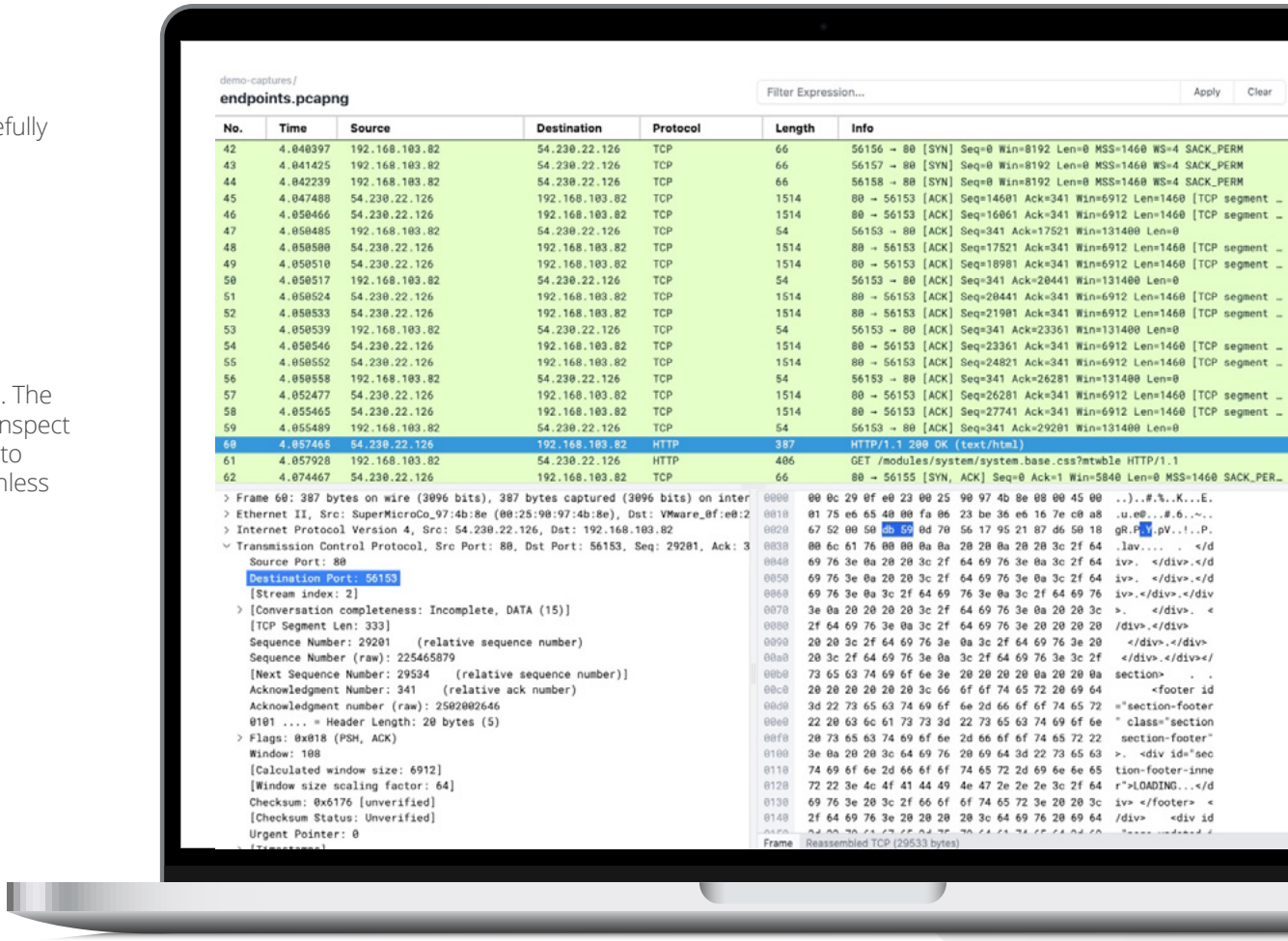
THIS DESIGN DELIVERS THREE STRATEGIC ADVANTAGES:

1) Reduced exposure surface
Fewer uncontrolled copies of sensitive data.

2) Simplified compliance posture
Easier to defend where evidence resides.

3) Stronger enterprise trust
Buyers see evidence handled within a controlled boundary.

In regulated industries and high-security environments, this is not a minor improvement. It is a meaningful architectural distinction.



Bringing it all together

Security platforms already capture packets. They already derive intelligence from them. They already depend on packet data to power AI detections and analytics. Competitive products elevate packet evidence from backend infrastructure to first-class investigative experience.

Alerts initiate investigations. AI analysis narrows uncertainty. But proof, the kind that withstands scrutiny from customers, executives, and auditors, lives at the packet level.

When packet evidence requires exporting data, switching tools, and reconstructing context manually, the investigative workflow fractures. The product feels incomplete at the exact moment trust is being earned.

Leading NDR platforms know that proof is in the packets

Packet Viewer exists to enable that shift without distracting your engineering team from its core mission. It embeds a mature, Wireshark-native packet analysis experience directly into your AI platform.

It allows you to design the investigative view your analysts need. It preserves context from alert to evidence. It keeps packet data inside your infrastructure. And it does all of this without turning packet tooling into a parallel product line you must build and maintain.

In markets where detection quality increasingly converges, workflow depth becomes the differentiator. The platforms that can move from insight to evidence in a single motion will stand apart from those that stop at summary.

Contact us to explore how Packet Viewer can help you elevate packet evidence to a first-class capability and strengthen your competitive AI position.

We believe that better networks make a better world. QA Cafe is dedicated to high-quality, user-focused test & analysis software with world class support.

Your success is our mission.

qa|cafe

www.qacafe.com