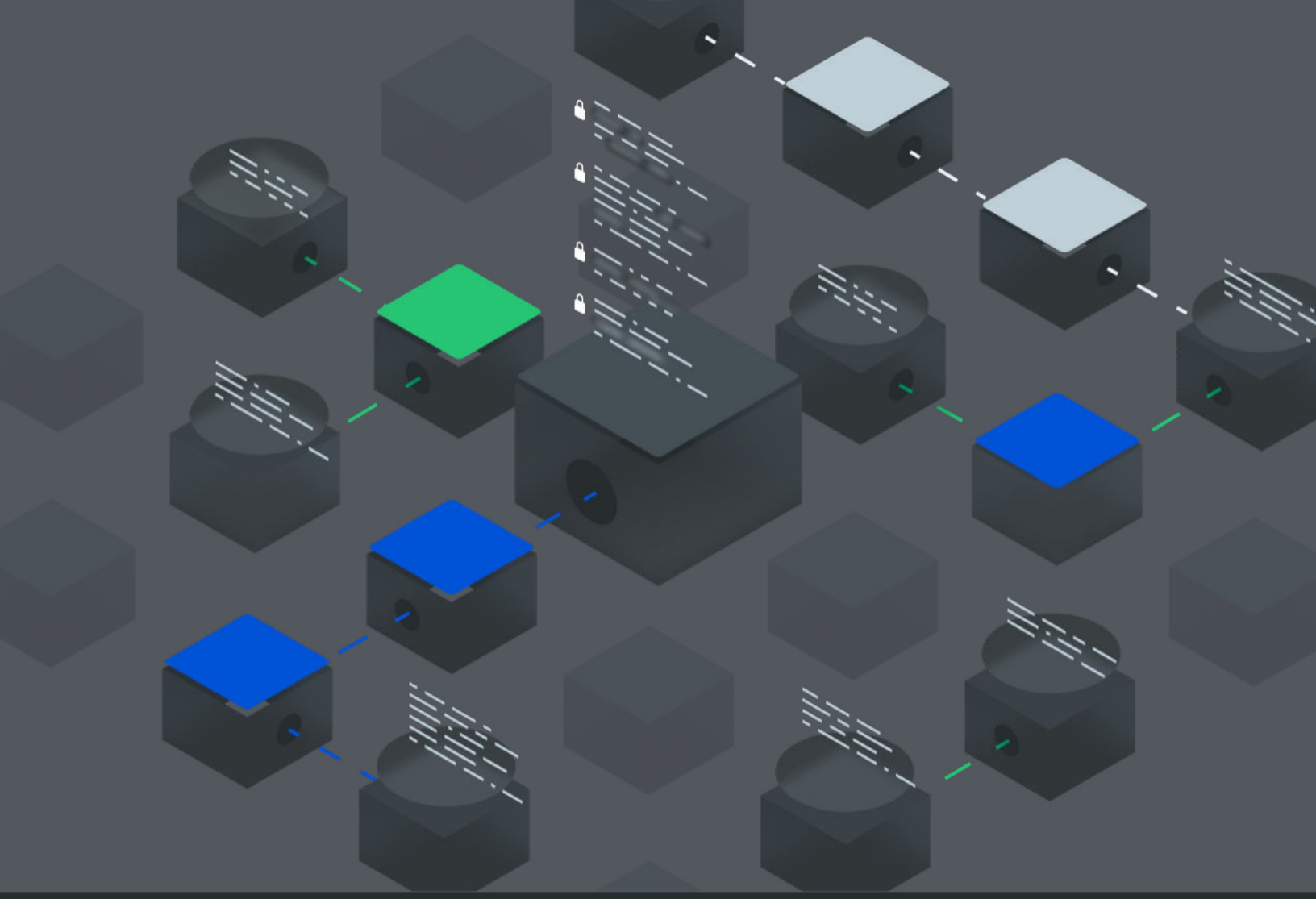

Traceable AI's Advanced AI/ML Detects and Solves Complex API Security Risks at Scale

TRACEABLE

Artificial intelligence (AI) and machine learning (ML) techniques show incredible promise for finding the proverbial needle in the haystack of complex data streams. However, this is not as straightforward for application programming interface (API) security as understanding language or finding cats in a photo. Novel data processing approaches, graph embeddings, and AI techniques are required to detect a new wave of API-based attacks that sometimes span hundreds of different services.

Traceable AI's approach to analytics-driven API security combines innovative approaches for reassembling data flow with unsupervised AI, graph data structures, and sophisticated anomaly detection techniques. These advanced technologies help detect attacks missed by traditional rule-based approaches or basic anomaly detection techniques such as API gateways and web applications firewalls (WAFs) alone.

Rules work up to a point

To understand how AI might help, it's worth exploring how today's security rule sets emerged in the first place. Over the years, enterprises have explored different approaches to protecting systems using simple rules or access control lists. These have gradually expanded into tools for protecting networking infrastructure at the IP, session, or application layers using intrusion detection systems (IDSs), intrusion prevention systems (IPSs), and WAFs.

These tools all look for and automatically block patterns of access to various hardware, software, and cloud services. Novice attackers tend to get stopped at the level of these products. But creative hackers have started picking apart weaknesses in the APIs that these devices cannot protect.

These more sophisticated attackers might explore more cunning and creative means that may span multiple APIs or operate outside of traditional security. For example, modern phishing attacks are getting better at mimicking the look of legitimate communications and tricking even some of the most guarded employees into signing into a credential harvesting server. Human creativity aligned with bots has raised the bar of new API attacks, making it harder to flag with traditional pattern-matching techniques.

Security companies are turning to anomaly detection techniques to address these new threats in addition to protecting against the older threat varieties. These tools are suitable for determining that some phenomena are odd – but not necessarily why. For example, the older tools might flag user requests coming from multiple physical regions in quick succession. But they could not necessarily tease out whether it was someone coming in from a wireless phone and laptop or part of a coordinated attack. So, the industry began exploring more sophisticated techniques – API security platforms – to categorize malicious behavior using AI and ML.

Challenges using AI to understand API

New AI techniques like transformers seem like they could be a natural fit for the task. Gigantic new AI language models like Google Bert and GPT 3.5 have demonstrated incredible results in interpreting speech, answering questions, and formulating responses. This has sparked some curiosity about how similar techniques could be used for analyzing other types of communication.

In theory, similar tools might help modern API security tools make sense of the language of API calls and responses across modern enterprises. In practice, a more nuanced approach is required. The big challenge security teams face is that the "language" spoken by APIs is a work in progress. Whereas the words, sentences, and grammar of human languages are stable, with APIs, the calls, responses, and permutations can change with every software update.

Another challenge is that most deep learning approaches use supervised learning models with data annotated by humans. For example, humans might indicate whether a picture has a cat using labels, which helps train algorithms to recognize cats. But in the case of API security, there is far less labeled data. Although it may be possible for security teams to label data after the fact, it is not practical to do this at scale. In addition, the resulting training data would not be relevant to another enterprise with a different API architecture.

Building another layer of **protection**

So Traceable AI took a step back to investigate how to build better algorithms using more automated techniques that can take advantage of data graphs to better characterize normal and abnormal behavior across API networks. This API security platform approach is already demonstrating promise for detecting many kinds of attacks more efficiently than traditional approaches. By creating a comprehensive view across multiple levels of abstraction, the Traceable algorithms can identify, flag, and block new attacks as soon as they happen.

For example, one basic type of API protection is called rate limiting. These services count the number of API calls over time and block new responses after a certain threshold. Rate limiting works for script-level attacks. However, sophisticated attackers know this and try to spread calls over time to avoid detection.

So, the notion of normal must be captured in different forms. Traceable has therefore developed a novel approach to look for abnormal behavior across five dimensions, including:

1. APIs
2. Parameters
3. Sequences of API calls or business logic
4. Sessions
5. Users

Let's walk through what this looks like in practice.

At the API level, Traceable identifies abnormal patterns of activity, such as an excessive number of service requests or rapid changes in location and IP address. Analyzing the APIs helps identify an abnormal number of access attempts. For example, the login API usually is only used once and a few times if someone mistypes or forgets their password. But calling the API 10-times in a row should raise a red flag since a hacker may be attempting to guess the password or user ID. Most rate throttling techniques look for these kinds of patterns but stop there.

At the parameter level, Traceable uses supervised learning techniques like one-class learning to characterize normal parameter combinations and identify rare events that may need further analysis across other levels. Additional analytics help identify an anomalous mix of parameters used in calling APIs. This can help identify issues like SQL injection and cross-site scripting attacks that can be executed within a single API call.

Traceable reassembles a series of API calls and parameter combinations into a sequence to reconstruct the business logic across services. For example, a user might login, add items to a cart, and log out. This can shed light on call sequence anomalies.

A session involves stringing together the user journey across different APIs. Session analytics supports business logic abuse detection. For example, it may flag cases where a user quickly logs in and logs out multiple times.

Lastly, Traceable curates a data graph to represent the concept of the user. This helps to detect signs of attacks at the user level, such as account takeover, and can detect data exfiltration attempts as well as privacy and compliance violations.

Putting it all together

Modern microservice architectures were designed for flexibility and scalability. One result is that microservices are decoupled from data stores, limiting the kinds of data they exchange. When a new API call comes in, the service does not know which user initiated the chain of calls for the current request. Traceable helps to recreate this chain of calls across APIs. It injects this data into an audit trail that tracks customer behavior in each session and chains together multiple sessions over time.

By piecing together multiple sessions over time into a coherent view of user behavior, Traceable uses behavioral anomaly detection to determine when something is wrong with the user. This helps block data exfiltration and fraud much earlier than approaches that only look at each API call or call sequence at the application level. For example, if an account is compromised, the hacker might begin slowly probing the system not to raise any rate throttling alerts.

Observability tools vendors are also starting to provide basic API protection capabilities. At their core, these tools apply observability to system metrics. This is suitable for identifying how many times an API is accessed, how parameters change, and how quickly new parameter combinations are tried. But may miss clues about malicious behavior hidden in the content of API messages. The API itself is a black box.

One of the challenges in analyzing complex patterns of interaction lies in accurately quantifying the interconnected web of links across users, sessions, sequences, parameters, and APIs. Traceable has pioneered novel techniques for representing the raw traces using graph embeddings. This supports a wider array of numerical tools for developers. In addition, it leads to significant performance improvements since the numerical computations involved are much faster than when using discrete graph algorithms.

Graph analytics techniques have been widely used to improve fraud detection algorithms by teasing apart the deeper connections between individuals, types of transactions, types of venues, and physical locations. Traceable AI has made some examples of these data sets available to the data science community to improve API analytics for the larger data science and security research community.

API security is a rapidly growing field as both hackers and security professionals improve their tools. Traceable believes that by shining a light on better techniques for organizing data and making sense of it, we can help the entire software industry move forward – and the API economy thrive.

About us.

Traceable was founded by third-time entrepreneur Jyoti Bansal and Sanjay Nagaraj. Bansal and Nagaraj saw the massive adoption of cloud-native architectures firsthand during their time at AppDynamics and founded Traceable as a result to protect applications from next-generation attacks.

Traceable applies the power of machine learning and distributed tracing to understand the DNA of the application, how it is changing, and where there are anomalies in order to detect and block threats, making businesses more secure and resilient.

traceable.ai

TRACEABLE_