

# Webinar

## APIs – das neue Einfallstor für Angreifer?

## Wie kann man APIs effektiv absichern?



# Bestandsaufnahme APIs



# 1061

Durchschnittliche Anzahl von Anwendungen pro Unternehmen

Darunter: "lebenswichtige" Anwendungen

Source: 2023 Connectivity Benchmark Report by MuleSoft In collaboration with Deloitte Digital

# APIs unter Sicherheitsgesichtspunkten



Mehr APIs jeden Tag



Mehr Angriffe  
Auf APIs



Mehr Datenvolumen  
durch APIs

Vorhandene Lösungen  
zum Schutz von  
Anwendungen sind  
zum Schutz von APIs  
nicht ausreichend

83%

des Datenvolumens im Web  
wird von APIs erzeugt<sup>2</sup>

Bis 2024 wird sich der Mißbrauch  
von APIs und Einbrüche über APIs  
nahezu

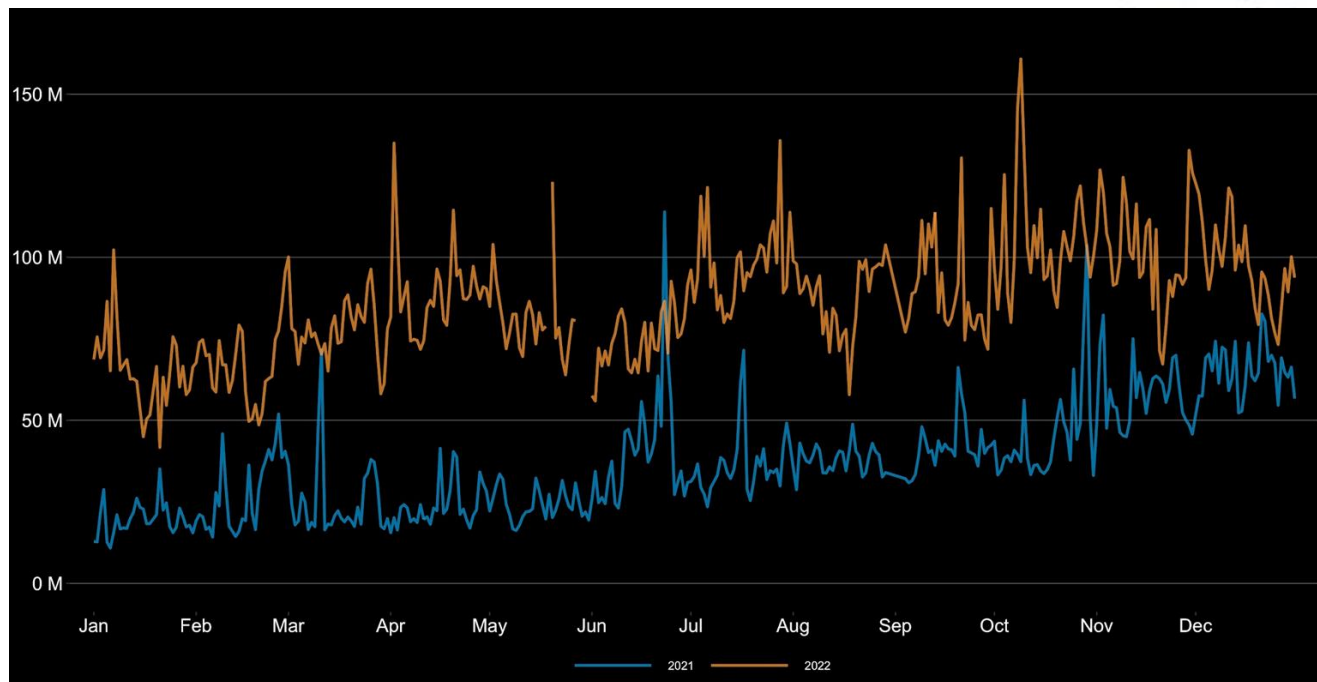
**verdoppeln<sup>1</sup>**

<sup>1</sup> Gartner: Top 10 Things Software Engineering Leaders Need to Know About APIs

<sup>2</sup> Akamai: Blog - API Discovery and Profiling -- Visibility to Protection

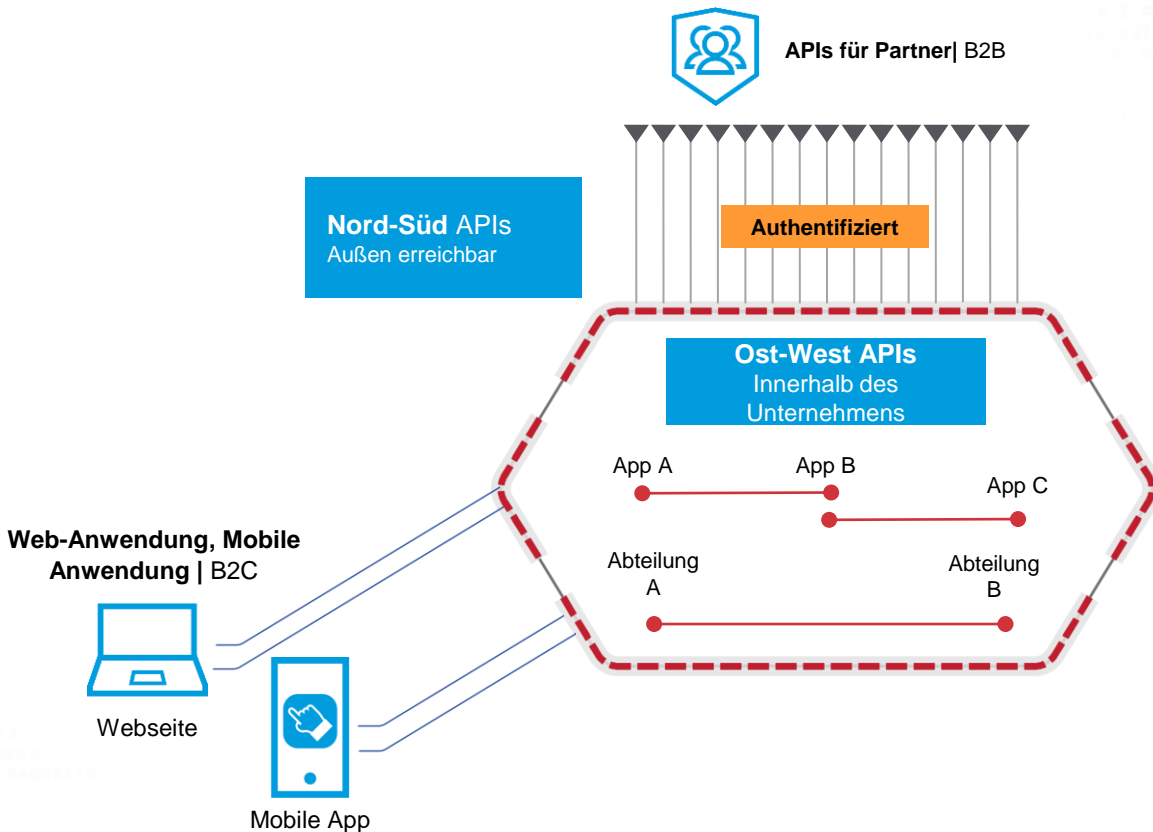
# Tägliche Angriffe auf APIs

Die Anzahl der Angriffe auf Web-Anwendungen und APIs hat sich mehr als verdoppelt.

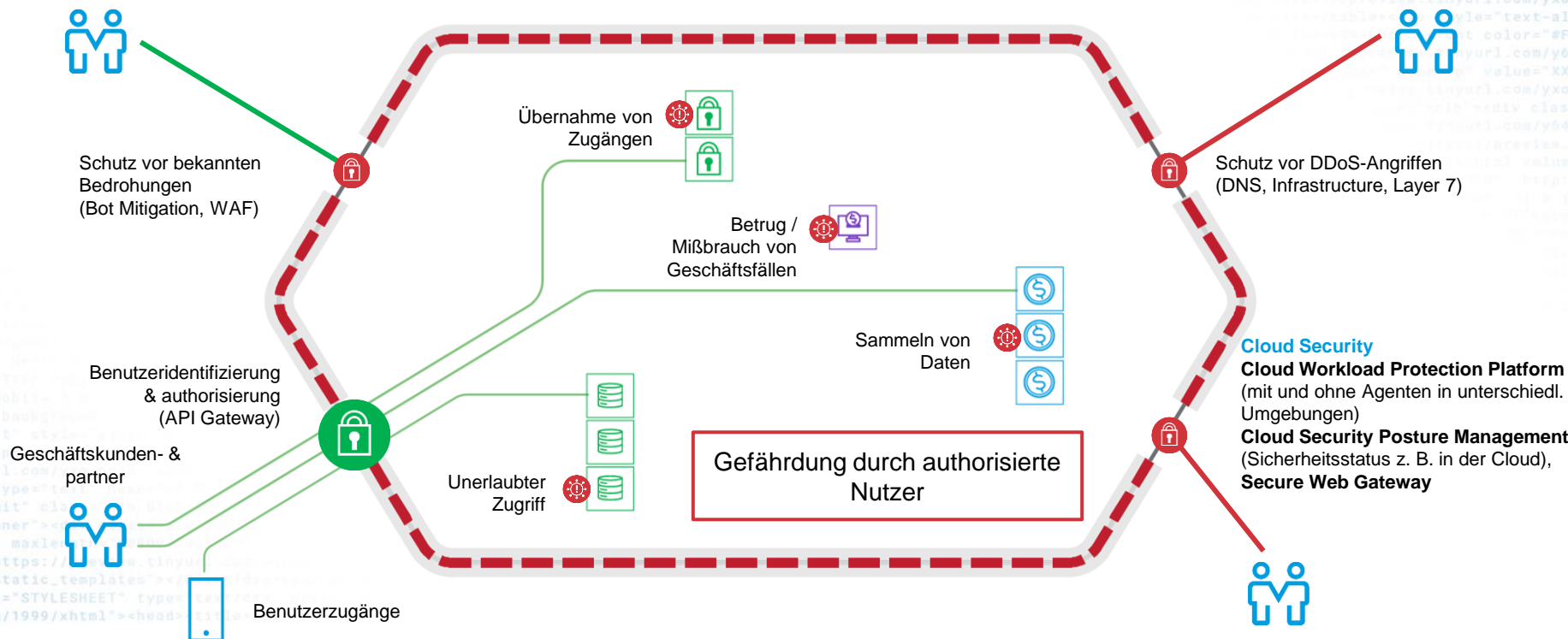




# Wie Apis verwendet werden



# Auch APIs mit beschränktem Zugang bieten eine große Angriffsfläche



# Umfrage 1

Wie genau ist Ihrer Meinung nach  
die Bestandsaufnahme der APIs,  
die derzeit in Ihrem  
Produktionsnetzwerk eingesetzt  
werden?





# Risiken bei der Verwendung von APIs

## OWASP Top 10 API Security Risks

# Risiken bei der Nutzung von APIs

## Versteckte APIs



Überblick über Ihre gesamte API-Infrastruktur:  
mit Einblick in mißbrauchte, veraltete, zum Test bestimmte und vergessene APIs

## APIs mit Schwachstellen



Verhindern von Angriffen auf bekannte OWASP Top 10 Schwachstellen und Fehlkonfigurationen

## Mißbrauch von APIs



Kontrolle wie Geschäftsfälle genutzt werden können, z. B. stoppen von Datensammlung oder Datendiebstahl über Erkennung von ungewöhnlichen Verhaltensmustern

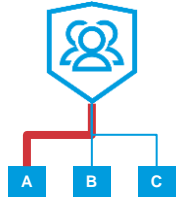
In Zukunft wichtig

Heute wichtig

# Sichtbarkeit | Keine Kontrolle auf Mißbrauch

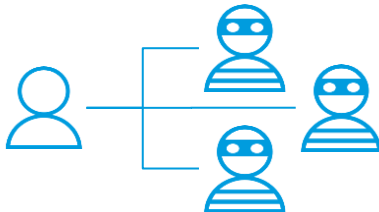
**Auch korrekt funktionierende APIs können mißbraucht werden**

**Übermäßige Nutzung durch  
Partner/Kunden/Angreifer**



**Ergebnis**  
Ausfall der API

**Absprachen von  
Benutzern im Spiel**



**Ergebnis**  
Verlust des Fairplays &  
geringere Attraktivität  
des Spiels

**Prämiensammlung für  
neu eröffnete Zugänge**



**Ergebnis**  
Zugänge werden zu  
betrügerischen  
Zwecken erzeugt

# Datengewinnung (“Scraping”) über APIs ist ein großes Risiko für Unternehmen

## Scraping: der Angriff auf APIs

- “unter dem Radar“ und langsam
- schwer zu bemerken
- Erkennung möglich über Verhaltenserkennung

# OWASP Top 10 API Security Risks (2023)

Rang	Risiko	Problembeschreibung
1	<u>API1:2023 - Broken Object Level Authorization (BOLA)</u>	Direkter Zugriff auf fremde Daten durch Ändern eines Bezeichners <b>Beispiel:</b> /api/Benutzer_ID/Kreditkartendaten/ => /api/Benutzer_ID_geraten/Kreditkartendaten
2	<u>API2:2023 - Broken Authentication</u>	Umgehen von Authentisierungsmechanismen <b>Beispiel:</b> Rücksetzen des Passwortes mit 4-stelligem Code ohne eine Beschränkung der Anzahl der Versuche
3	<u>API3:2023 - Broken Object Property Level Authorization</u>	Zugriff auf zu viele und/oder schützenswerte Informationen <b>Beispiel:</b> Token zum Zurücksetzen von Zugängen in Antworten auf Anfragen
4	<u>API4:2023 - Unrestricted Resource Consumption</u>	Keine Beschränkung der Anzahl der Anfragen Beispiel: Brute Force-Angriffe auf Kode zum Zurücksetzen des Passworts (6 Ziffern, 200 Versuche pro IP in 10 Minuten)
5	<u>API5:2023 - Broken Function Level Authorization (BFLA)</u>	Benutzer erhalten ungewollten Zugriff auf <b>Funktionen</b> <b>Beispiel:</b> Löschen von anderen Benutzerzugängen ("DELETE /api/user/id"), Zugriff auf Premiumfunktionen



# OWASP Top 10 API Security Risks (2023)

Rang	Risiko	Problembeschreibung
6	<u>API6:2023 - Unrestricted Access to Sensitive Business Flows</u>	Benutzer können Funktionen unbeschränkt nutzen <b>Beispiel:</b> Kauf einer Eintrittskarte oder unberechtigte Preisermäßigungen beim Erstellen von neuen Benutzerzugängen
7	<u>API7:2023 - Server Side Request Forgery</u>	Angreifer können Anwendungen in Unternehmen hinter einer Firewall erreichen, z. B. wenn ein Eingabefeld für eine manipulierte URL genutzt wird.
8	<u>API8:2023 - Security Misconfiguration</u>	Fehler bei der Konfiguration der Sicherheitseinstellungen von APIs Fehler in der Software, die die API zur Verfügung stellt <b>Beispiel:</b> API-Endpunkte, die ein Löschen von Daten ermöglichen oder Logs, die schützenswerte Informationen enthalten
9	<u>API9:2023 - Improper Inventory Management</u>	Angreifer können unbekannte & ungesicherte APIs für Angriffe nutzen. Durch nicht mehr genutzte, aber erreichbare APIs, steigt die Angriffsfläche für die Organisation/das Unternehmen.
10	<u>API10:2023 - Unsafe Consumption of APIs</u>	Nutzung von Drittanbieter APIs, die keine ausreichenden Sicherheitsmechanismen einsetzen (z. B. "Filtern von Eingabedaten")



# OWASP Top 10 API Security Risks (2023) □ Gegenmaßnahmen

Rang	Risiko	Gegenmaßnahmen
1	<u>API1:2023 - Broken Object Level Authorization (BOLA)</u>	<ul style="list-style-type: none"><li>- Überprüfen der Berechtigung über Richtlinien und Vererbung</li><li>- keine Benutzer-IDs in API-Anfragen</li><li>- Nutzung von Sitzungs-IDs oder Token</li><li>- nicht ratbare-IDs, zufällig gewählte IDs</li></ul>
2	<u>API2:2023 - Broken Authentication</u>	<ul style="list-style-type: none"><li>- Zugriffssperre nach zu vielen Fehlversuchen</li><li>- sichere Länge von Einmalpasswörter</li><li>- sichere Rücksetzung des Passworts</li><li>- Multifaktor-Authentifizierung</li></ul>
3	<u>API3:2023 - Broken Object Property Level Authorization</u>	<ul style="list-style-type: none"><li>- Filterung der Daten, die die API liefert</li><li>- Beschränkung der Ausgabe von schützenswerten Informationen</li><li>- (Schwachstellen-)Test auf Herausgabe schützenswerter Informationen</li></ul>
4	<u>API4:2023 - Unrestricted Resource Consumption</u>	<ul style="list-style-type: none"><li>- Beschränkung der Anzahl der Zugriffe</li><li>- Beschänkung der Größe der empfangenen Daten</li><li>- Bestimmung der erlaubten Rate für den Zugriff</li></ul>
5	<u>API5:2023 - Broken Function Level Authorization</u>	<ul style="list-style-type: none"><li>- Vermeidung der Authorisierung für Funktionen</li><li>- Blockieren aller Zugriffe als Standard</li><li>- Erlaubnis des Zugiffs nur für privilegiert Benutzer</li><li>- regelmäßige Tests der Zugriffsmöglichkeiten</li></ul>

# OWASP Top 10 API Security Risks (2023) □ Gegenmaßnahmen

Rang	Risiko	Gegenmaßnahmen
6	<u>API6:2023 - Unrestricted Access to Sensitive Business Flows</u>	<ul style="list-style-type: none"><li>- Entwickler sollten verstehen, wie die API im Geschäftsverkehr genutzt werden sollte und Mißbrauch verhindern, z. B. durch Rate Controls</li></ul>
7	<u>API7:2023 – Server-Side Request Forgery</u>	<ul style="list-style-type: none"><li>- Einsatz einer API-Sicherheitslösung, die ungewöhnliche Pfade in API-Anfragen erkennt</li><li>- Einsatz einer API-Sicherheitslösung, die auf normale API-Anfragen trainiert wird und Abweichungen erkennt</li></ul>
8	<u>API8:2023 - Security Misconfiguration</u>	<ul style="list-style-type: none"><li>- Einführung und Kontrolle von Härten und Fehlerbehebung in APIs</li><li>- Abschalten aller unnötigen Komponenten und Funktionen</li><li>- Beschränkung der Zugriffsrechte ("nur was nötig ist")</li></ul>
9	<u>API9:2023 - Improper Inventory Management</u>	<ul style="list-style-type: none"><li>- Bestandsaufnahme über alle vorhandenen APIs</li><li>- vollständige Dokumentation der APIs</li><li>- Abschalten nicht mehr gebrauchter APIs</li></ul>
10	<u>API10:2023 - Unsafe Consumption of APIs</u>	<ul style="list-style-type: none"><li>- Verwenden des Ansatzes: "Traue niemanden": Überprüfung der Daten, die von Drittanbietern geliefert werden</li></ul>

# Akamai API-Security schützt gegen die 10 wichtigsten API- Sicherheitsrisiken der OWASP

<https://www.akamai.com/de/resources/checklist/owasp-api-top-10>

## AKAMAI-CHECKLISTE

## OWASP Top 10 API-Sicherheit

APIs haben sich zum Standard bei Aufbau und Verbindung moderner Anwendungen entwickelt, insbesondere angesichts des zunehmenden Umstiegs auf Microservices-Architekturen. Aus diesem Grund ist es wichtig, Ihr Unternehmen vor den häufigsten API-Sicherheitsrisiken zu schützen, die durch das Open Worldwide Application Security Project (OWASP) ermittelt wurden. Werfen Sie einen Blick auf die aktuelle Liste für 2023, um besser über Methoden zum Schutz Ihrer APIs informiert zu sein.

### Abdeckung der OWASP API Top 10 durch Akamai

- ☒ **API1:2023 – Fehlerhafte Autorisierung auf Objektebene:** Sicherheitslücken durch fehlerhafte Autorisierung auf Objektebene (Broken Object Level Authorization – BOLA) entstehen, wenn die Autorisierung eines Clients nicht ordnungsgemäß für den Zugriff auf spezifische Objekt-IDs validiert wird.
- ☒ **API2:2023 – Fehlerhafte Authentifizierung:** Fehlerhafte Authentifizierung bezieht sich auf weitreichende Sicherheitslücken im Authentifizierungsprozess, durch die das System Angreifern ausgesetzt ist. Diese können Schwachstellen ausnutzen, um den API-Objektschutz zu gefährden.
- ☒ **API3:2023 – Fehlerhafte Autorisierung auf Objekteigenschaftsebene:** Fehlerhafte Autorisierung auf Objekteigenschaftsebene (Broken Object Property Level Authorization – BOPLA) ist ein Sicherheitsfehler, bei dem ein API-Endpunkt unnötig mehr Dateneigenschaften offenlegt, als für seine Funktion erforderlich ist, und so das Prinzip der geringstmöglichen Berechtigungen vernachlässigt wird.
- ☒ **API4:2023 – Uneingeschränkte Ressourcennutzung:** Sicherheitslücken dieser Art werden manchmal als API-Ressourcenüberlastung bezeichnet. Dabei werden die Anzahl der Anforderungen oder das Datenvolumen, das die API innerhalb eines bestimmten Zeitraums bearbeiten kann, durch die APIs nicht begrenzt.
- ☒ **API5:2023 – Fehlerhafte Autorisierung auf Funktionsebene:** Fehlerhafte Autorisierung auf Funktionsebene (Broken Function Level Authorization – BFLA) kann auftreten, wenn Zugriffskontrollmodelle für API-Endpunkte falsch implementiert sind.
- ☒ **API6:2023 – Unbeschränkter Zugriff auf sensible Geschäftsabläufe:** Dieses Risiko entsteht, wenn eine API kritische Vorgänge wie Geschäftslogik ohne ausreichende Zugriffskontrolle offenlegt.
- ☒ **API7:2023 – Serverseitig manipulierte Anforderungen:** Serverseitig manipulierte Anforderungen (Server Side Request Forgery – SSRF) ermöglichen es einem Angreifer, die serverseitige Anwendung dazu zu veranlassen, HTTPS-Anfragen an eine beliebige Domain seiner Wahl zu senden.
- ☒ **API8:2023 – Fehlerhafte Sicherheitskonfiguration:** Dies bezieht sich auf die unsachgemäße Einrichtung von Sicherheitskontrollen, die ein System anfällig für Angriffe machen kann.
- ☒ **API9:2023 – Fehlerhafte Bestandsverwaltung:** Dies stellt für jedes Unternehmen, das APIs verwaltet, eine Herausforderung dar. API-Sicherheitslösungen können bekannte APIs schützen. Unbekannte und auch veraltete APIs sind aber möglicherweise nicht gepatcht und daher anfällig für Angriffe.
- ☒ **API10:2023 – Unsichere Nutzung von APIs:** Dies bezieht sich auf die Risiken, die mit der Verwendung von Drittanbieter-APIs verbunden sind, wenn keine angemessenen Sicherheitsmaßnahmen ergriffen werden.

### Zusammenarbeit mit uns

Unternehmen und Ihre Sicherheitsanbieter müssen eng zusammenarbeiten und Menschen, Prozesse und Technologien zusammenbringen, um effektiven Schutz vor den in den OWASP API Security Top 10 beschriebenen Sicherheitsrisiken zu gewährleisten.

### Über Akamai

Akamai bietet branchenführende Sicherheitslösungen, erfahrene Experten und die Akamai x Connected Cloud mit Einblicken in Millionen von Webanwendungsangriffen, Milliarden von Bot-Anfragen und Billionen von API-Anfragen pro Tag. Die Sicherheitslösungen für Webanwendungen und APIs von Akamai schützen Ihr Unternehmen vor den fortschrittlichsten Formen von Webanwendungen, DDoS (Distributed Denial of Service) und API-basierten Angriffen.

# Wie Sicherheit von APIs aussehen sollte

In welcher Phase befindet sich Ihr Unternehmen?

Phase 1	Phase 2	Phase 3	Phase 4	Phase 5	Phase 6
Bestandsaufnahme für APIs	Finden von APIs	Risikobewertung	Verhaltens-erkennung	Reaktion	Überwachung & Erk. neuer Bedroh.
<p>Werden Aktivitäten über APIs geloggt?</p> <p>Reichen diese Logs aus?</p> <p>Wie wird z. B. mit personenbezogenen Daten umgegangen?</p>	<p>Kennen Sie alle Ihre Microservices?</p> <p>Kennen Sie alle Ihre APIs?</p>	<p>Wie sieht Ihre Gefährdungslage aus?</p> <ul style="list-style-type: none"> <li>• Fehl-konfigurationen?</li> <li>• Fehler?</li> <li>• Dokumentation?</li> <li>• Kritische Daten?</li> </ul>	<p>Sind Sie in der Lage Mißbrauch von APIs oder Geschäftsfällen zu erkennen?</p> <p>Können Sie die Nutzer Ihrer APIs erkennen?</p>	<p>Einrichten von automatischen Reaktionen auf Vorfälle?</p> <p>Können die Reaktionen angepasst werden?</p>	<p>Sind Sie in der Lage Angriffe in Logs zu finden?</p> <p>Können Sie Bedrohungen in den APIs aufdecken?</p>
Nutzen Sie vorhandene Daten. Spezielle Sensoren für APIs sind nicht nötig.	In diesem Phase sollten alle APIs gefunden werden.	Überprüfung Ihrer gesamten API-Landschaft	Verhaltens-erkennung ist nur mit Datensammlung und mit Hilfe eines SaaS-Dienstes möglich	Eine offene Plattform um Reaktionen auf Vorfälle festzulegen hilft hier.	Datensammlung mit Hilfe eines SaaS-Dienstes nötig

# Umfrage 2

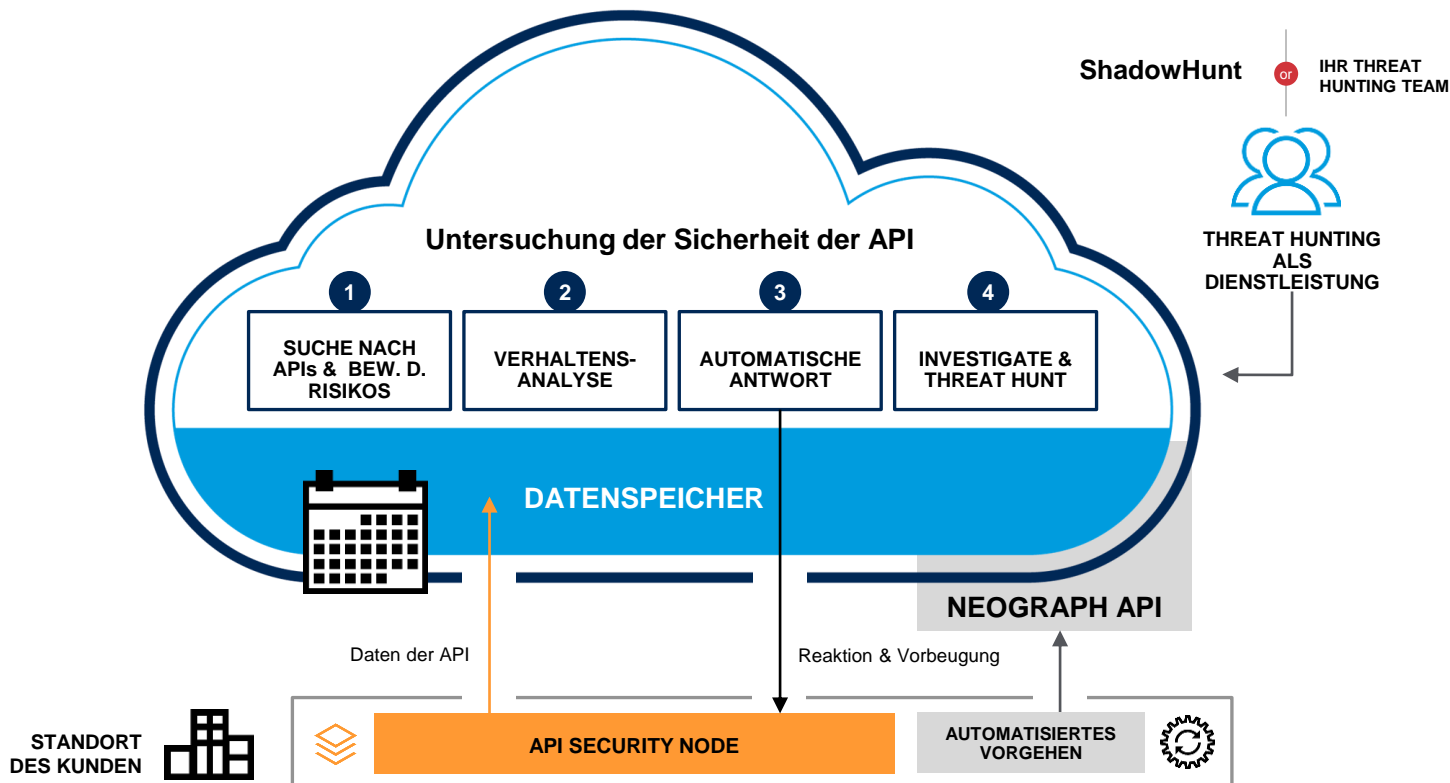
Welche Technologien/Tools  
werden derzeit zum Schutz Ihrer  
APIs verwendet?

# Eine Sicherheitslösung für APIs

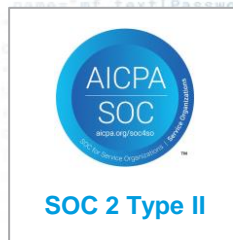




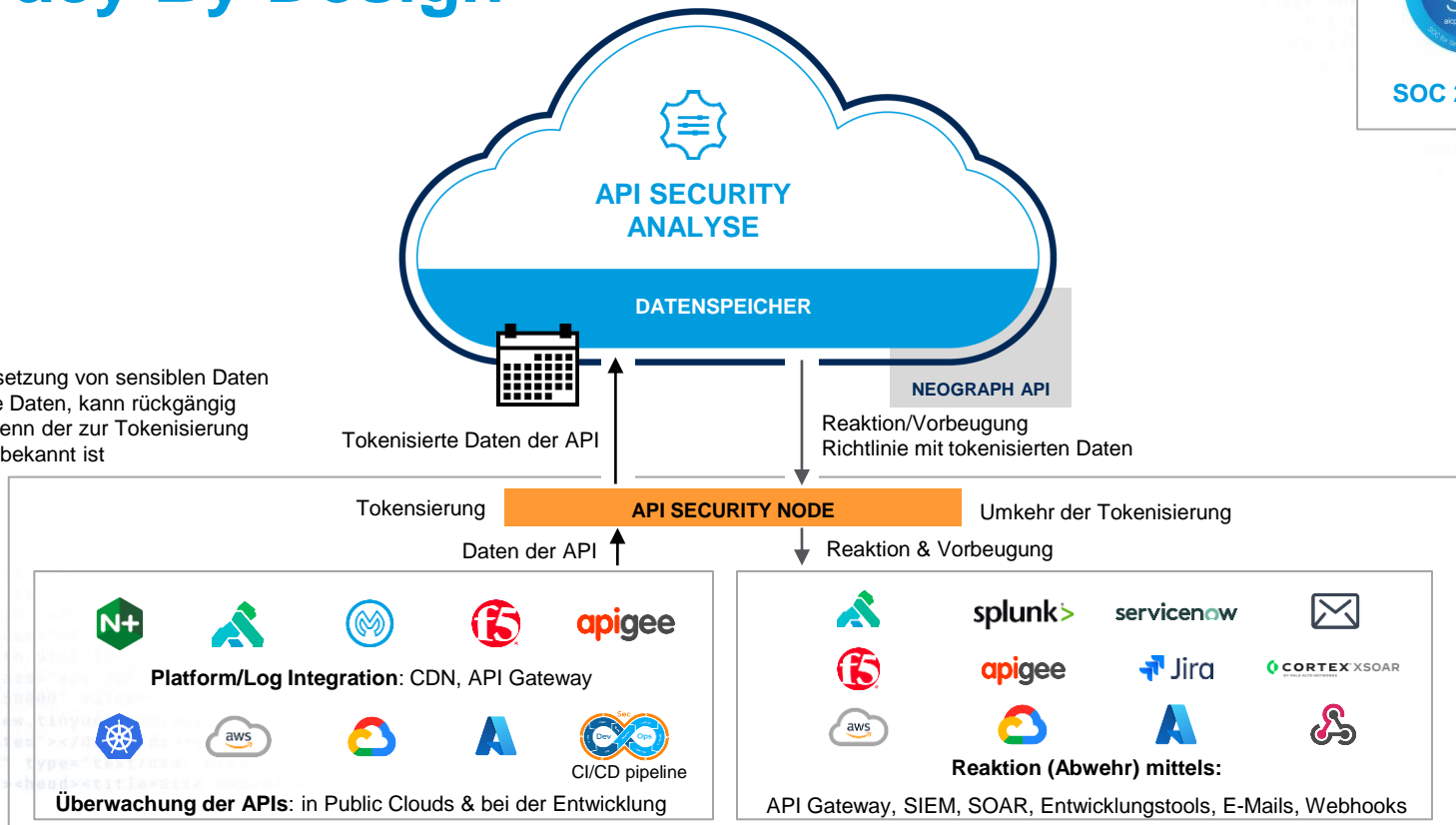
# Schutz für APIs



# Privacy By Design



**Tokenisierung:** Ersetzung von sensiblen Daten durch nicht sensible Daten, kann rückgängig gemacht werden, wenn der zur Tokenisierung genutzte Schlüssel bekannt ist



# Umfrage 3

Welche Technologien/Tools  
fehlen Ihnen aktuell zum Schutz  
ihrer APIs?

# Demonstration



# Fragen & Antworten







**Akamai**