12 THINGS
DEVOPS & IT SECURITY MUST DO TO PROTECT APIS
INTRODUCTION

API-based attacks are a new breed of sophisticated intrusions—and they’re all around us. Facebook, Google, Verizon, USPS and T-Mobile are just some of the enterprises that have recently reported attempts to steal data and access applications, and in many cases, the attacks weren’t discovered until long after the fact.

One primary reason behind this surge is that there typically isn’t much standing between a hacker and an API. Bad actors commonly breach APIs via a vulnerability discovered by reverse engineering the API, and with the extensive use of automation. No solution based on attack signatures or rules can detect these attacks, nor is there an easy way to test the large number of exploitable use case permutations.

To combat these attacks, access control is not enough. It’s also no longer solely about countering SQL injection or XSS attack attempts. It’s about guarding against API reverse engineering that shows hackers how to best get to data and applications. It’s about protecting APIs against those who log in with stolen credentials. It’s about thwarting the theft of data and private information, and preventing hackers from disabling APIs via finely tuned, low-volume, targeted DDoS attacks.
**THESE TOP 12 BEST PRACTICES ARE A MUST FOR PROTECTING YOUR API INFRASTRUCTURES AGAINST HACKING AND ABUSES.**

1. Fund and assemble a team to explore, identify and deploy solutions and processes that address the broader cybersecurity posture of your organization’s APIs.
   
   Leadership means recognizing the issue upfront, delivering guidance on proper API security design techniques, and deploying solutions that can track API activity and block threats.

2. Test APIs and data stores for security.
   
   And as you do so, “think like a hacker.” Make sure the API developer is involved in penetration testing efforts as they’ll be aware of potential weak spots and vulnerabilities. For example, API security was bypassed by simply changing the payload type in the breach of LocationSmart.

3. Adhere to a continuous security mindset during development.
   
   It is good practice to insert API security expertise within DevOps teams and to constantly remind developers of the need to properly authorize each attempt to access data, among a range of other best practices.

4. Conduct automated security scans, tests and monitoring.
   
   It’s important to scan code for vulnerabilities outlined in security frameworks like the OWASP Top 10 before going to production. Otherwise, easy-to-remedy attack vectors like weak authentication, session management and security misconfigurations could be the cause of sensitive data exposure.

5. Deploy a strong authentication and authorization system in all tiers.
   
   Where appropriate, security measures like multi-factor authentication, continuous authentication and authorization, and properly validating tokens/cookies can go a long way.

6. Utilize flow control and TLS encryption at all times.
   
   This prevents DoS attacks and allows data to be transmitted between parties securely so that it cannot be intercepted, eavesdropped on or tampered with.
Prevent app servers from sending error messages with system traces.
Hackers are good at forcing errors when probing APIs with the objective of receiving an error message that includes debug traces with IP addresses, system names, etc.

Do NOT register internal API names in a public DNS.
It's best to keep all API names internal and to map them into external names that are used with DNS servers. This practice also has the added benefit of enabling you to change internal API names without having to change DNS entries as well.

Track ALL APIs.
Tools can be used to auto-discover APIs so that those in production, and also those used for testing or maintained for backwards compatibility, are never forgotten. All APIs should be monitored by your security team.

Perform periodic reviews of API activity to identify abnormalities.
You need complete visibility into every access attempt and session so that you're always aware of API activity. Lack of visibility is one of the reasons why so many breaches go undetected after months of ongoing data theft.

Treat all APIs as if they were externally facing.
This includes internal APIs and those in labs, which can be the root causes of data breaches. These APIs are often forgotten by developers after deployment, and sometimes DevOps and security teams are never made aware of their presence in the first place. All APIs must be protected and secured in a standard fashion, and activity should be tracked with an audit trail to identify abnormal patterns.

Leverage new technology to give your enterprise a fighting chance.
Bad actors are using AI in their attempts to breach your systems. Your enterprise must look beyond traditional tools to find new ways to detect and stop attacks on APIs before they have an impact. For example, PingIntelligence for APIs applies AI models to continuously inspect and report on all API activity while detecting and stopping attacks that use APIs to compromise data and applications.

These days, one of your biggest threats is ignoring the need to do more to secure your APIs. To learn more about how to protect your APIs against cyber attacks and data breaches, download the analyst report “Gartner: How to Build an Effective API Security Strategy.”