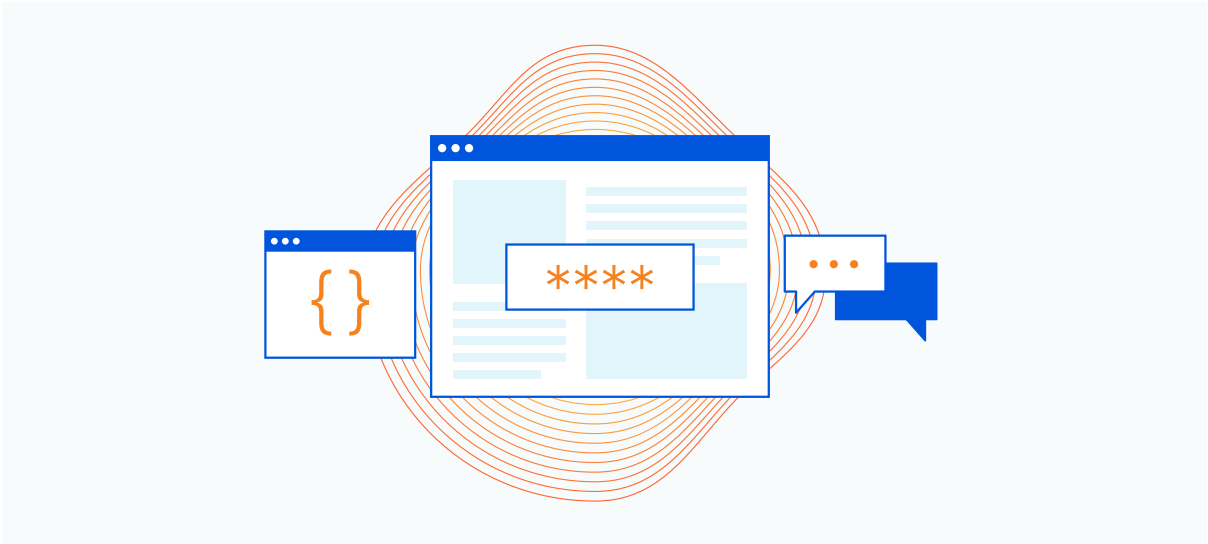
ARTIGO TÉCNICO

CLOUDFLARE

Um guia de segurança para APIs

+55 (11) 3230 4523 | enterprise@cloudflare.com | www.cloudflare.com/pt-br/

As APIs movem o mundo (dos aplicativos)



Todos nós já sabemos que as APIs, interfaces de programação de aplicativos, movem o mundo. Mais precisamente, elas permitem que os diferentes aplicativos modernos se comuniquem entre si. Aplicativos web ou móveis podem acessar um back-end onde os dados são armazenados e processados. As APIs podem ser públicas, permitindo que aplicativos de diferentes empresas se comuniquem, ou privadas, o que é comum, onde os aplicativos internos se integram para atender aos objetivos da empresa.

O resultado? Aplicativos mais robustos, plenos, para sites e dispositivos móveis, com mais funcionalidades e dados mais diversos.

Por exemplo, em vez de criar seus próprios serviços de pagamento do zero, as empresas de compartilhamento de viagens podem adicioná-lo por meio das APIs das empresas de pagamento. Outro exemplo são os sites agregadores de voos. Para nos mostrar horários de voos, preços, destinos e tudo o mais que precisamos saber sobre uma passagem de avião, eles se conectam por meio de chamadas de API aos bancos de dados das companhias aéreas para obter os dados corretos e exibi-los em nossa página de resultados de pesquisa do agregador.

A importância das APIs está crescendo, com mais e mais empresas se descrevendo como “API-first”. Em alguns casos, o produto real de uma empresa é uma API com um modelo de negócios centrado no consumo dela. Por exemplo, se uma empresa que fornece dados climáticos tornou sua API seu produto, outras empresas que desejam informações meteorológicas pagarão uma tarifa mensal pelo acesso à API. Em muitos outros casos, dada a expectativa de que um aplicativo deve interagir com outros aplicativos, as APIs são projetadas ao lado ou mesmo antes do código que fornece a funcionalidade real do produto – não adicionadas ao final do desenvolvimento.

As empresas dedicam tempo e esforço para elaborar deliberadamente sua abordagem de API para considerar como ela pode expor os dados certos, tornando-se fundamental para a receita e os modelos de negócios.

No entanto, criar APIs perfeitas é difícil porque, como qualquer software, vulnerabilidades ocorrerão, levando aos desafios de segurança que discutiremos neste documento.

Basta dizer que as APIs estão em toda parte e só ganharão força nos próximos anos – e elas devem ser protegidas. Por esse motivo, examinaremos os ataques às API e os aspectos de defesa em profundidade, ao pensar na segurança da API organizacional.

O momento da API em números

50%

do tráfego que flui pela Cloudflare é de APIs

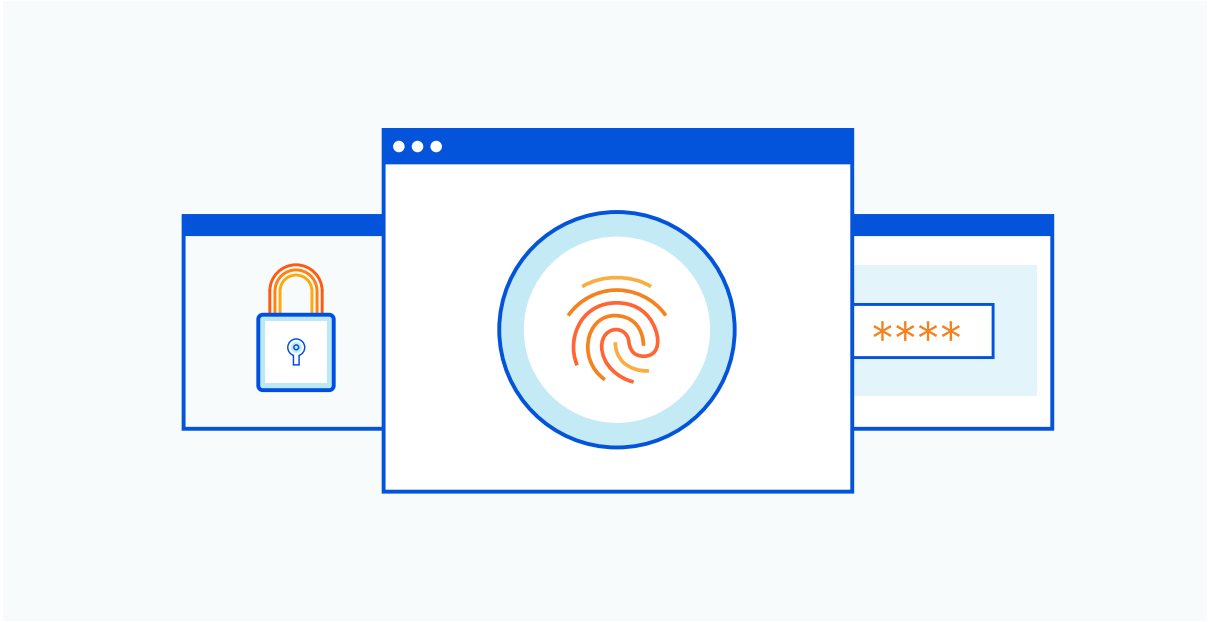
61%

de aumento no tráfego de APIs em termos anuais

A Programmable Web[1] observou que há mais de 24 mil APIs publicadas e bem conhecidas. No entanto, a maioria é privada e conecta aplicativos internos. A estimativa para número de APIs privadas está na casa dos milhões.

E quando dizemos que as APIs estão ganhando força, a Cloudflare é uma testemunha em primeira mão de seu crescimento. De acordo com os dados do Radar da Cloudflare do primeiro semestre de 2021, aproximadamente metade do tráfego na Rede da Cloudflare estava relacionado às APIs. Além disso, o tráfego aumentou 61% de 2020 para 2021.

Como as APIs expõem dados importantes, começamos a perceber que elas representam uma enorme nova superfície de ataque, que deve ser protegida. E como sabemos disso? Houve vários ataques relevantes a APIs nos últimos anos.



[1]https://www.programmableweb.com/apis/directory

Uma superfície de ataque cada vez maior

Sabemos que as APIs estão em todos os lugares e são fundamentais para o sucesso de uma empresa moderna. Elas expõem a lógica dos aplicativos e podem compartilhar dados sensíveis com outros aplicativos.

No entanto, não é surpresa que os invasores saibam disso e tenham a intenção de explorar essa superfície de ataque cada vez maior nas empresas.

Talvez a Gartner esteja certa quando afirma[2] que até 2022, os abusos de APIs passarão de um vetor de ataque pouco frequente para o mais frequente, resultando em invasão de dados em aplicativos web corporativos.

Ainda vamos ver se as APIs se tornarão o vetor de ataque mais visado , mas está claro que elas continuarão na mira dos invasores.

"Continuarão" porque o mundo já viu invasões de grande importância resultantes de uma segurança de API fraca ou um desenvolvimento de API que não levou em conta a segurança.

A T-Mobile foi vítima de um ataque à API em 2017, quando 15 milhões de clientes que compraram novos dispositivos ou criaram contas na T-Mobile tiveram informações expostas, como nome, endereço, data de nascimento, números de seguro social, carteira de habilitação e passaporte. A Vice reportou que o ataque foi realizado por meio do ajuste do parâmetro de número de telefone na chamada à API. O número de telefone de qualquer usuário podia ser consultado e a API da T-Mobile enviava respostas que incluíam dados sensíveis da conta da pessoa cujo número de telefone era consultado.

Outra invasão relacionada à API atingiu a USPS quando uma API que suportava o rastreamento de pacotes em tempo real não tinha autorização básica. Quando um usuário estava logado, ele podia consultar as informações da conta de qualquer outro usuário, por meio do uso de parâmetros de pesquisa curinga que capturavam todos os registros do conjunto de dados. Isso colocou em risco os dados de 60 milhões de correntistas da USPS.

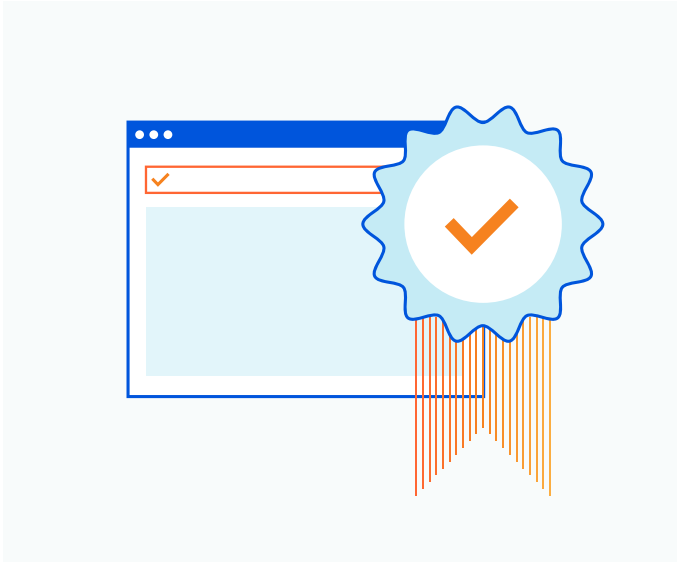
Em 2019, a JustDial, um grande mecanismo de pesquisa local na Índia, de fato vazou todos os dados de clientes quando deixou expostas as versões antigas das APIs que tinham sido substituídas por outras mais novas. E ficou ainda pior – não havia nenhuma autenticação implementada, ou seja, qualquer pessoa poderia chamar as APIs e raspar dados do servidor de produção. Em outras palavras, não era necessária nenhuma técnica avançada para acessar os dados dos usuários.

O Facebook, apesar da liderança com GraphQL, também sofreu diversas invasões a APIs. Por exemplo, no fim de 2019, o banco de dados do Facebook foi atacado e nomes, números de telefone e identidades de usuários de mais de 260 milhões de pessoas foram colocados em risco.

Há uma longa lista de empresas que passaram por problemas com a segurança das APIs. E também há vários motivos para isso. Primeiro, as vulnerabilidades subjacentes da API, devido ao design pouco seguro, abrem portas para ataques. Além disso, até hoje, as empresas ainda não têm ferramentas de segurança exclusivas para APIs. Talvez elas usem ferramentas de segurança para a web, como WAFs ou Rate limiting, mas elas foram criadas para proteger aplicativos e não APIs. Isso pode criar desafios, como falsos positivos, e deixa clara a necessidade de uma segurança para APIs projetada para um tráfego que é, em grande parte, automatizado.

[2]Fonte: Gartner: “API Security: What You Need to Do to Protect Your APIs”, março de 2021

Remix! O novo Top 10 do OWASP para APIs



O lado bom de tudo isso é que a Fundação OWASP, uma organização que há muito tempo trabalha para aumentar a segurança de aplicativos, agora faz parte dessa iniciativa. O OWASP é conhecido pelo Top 10 de riscos à segurança de aplicativos web, e agora publicou o Top 10 de segurança para APIs, uma lista dos principais riscos e vulnerabilidades de segurança para APIs.

A verdade é que, nossa antiga preocupação sobre a segurança de aplicativos também se aplica à criação e proteção de APIs.

Para começar, toda organização que é API-first deve considerar a segurança logo de início, no projeto das APIs. Vamos examinar alguns dos ataques que acabamos de mencionar e o risco de segurança do OWASP explorado.

Top 10 do OWASP para APIs

1. Falha de autorização no nível do objeto
2. Falha de autenticação de usuário
3. Exposição excessiva de dados
4. Falta de recursos e Rate Limiting
5. Falha de autorização no nível da função
6. Atribuição em massa
7. Erro de configuração de segurança
8. Injeção
9. Gestão de ativos inadequada
10. Registros e monitoramento insuficientes

Principais desafios de segurança para APIs

1. Falha na autenticação e na autorização

Vamos olhar mais de perto alguns dos principais riscos do OWASP para APIs explorados pelos ataques acima, começando com autenticação e autorização.

A JustDial sucumbiu por causa de falhas de autenticação nos endpoints, que permitiam que qualquer pessoa realizasse chamadas para eles. Com a autenticação implementada, somente chamadas à API com o certificado TLS, chaves de API, tokens da web etc. corretos podem fazer solicitações, o que reduz drasticamente o risco de segurança para a API.

Falando sobre o primeiro lugar na lista do OWASP, muitos ataques à API exploram a autorização fraca, com falhas ou não existente, como aconteceu com a USPS e a T-Mobile. Falhas de autorização no nível do objeto são comuns - quando os endpoints da API são explorados substituindo o número de ID de um objeto, eles têm autorização para acessar a ID de algo que eles não estão autorizados a acessar. Muitas vezes, a simples alteração da ID do objeto em uma solicitação dá acesso não autorizado a dados.

O caminho da API e os parâmetros de consulta incluem a ID do recurso acessado:

Chamada autorizada:

`GET api.greatsampleapis.com/v1/users/235` , em que 235 é a ID do usuário.

Chamadas à API manipuladas podem conseguir acesso não autorizado ao alterar 235 para 236, ou seja, ajustar o identificador do objeto, nesse caso, userID, para acessar dados do usuário 236.

```
GET api.greatsampleapis.com/v1/users/236
```

Se não houver controles de autorização, esse ataque pode ser bem-sucedido. Os desenvolvedores devem modelar seu endpoint contra ameaças para garantir que os ataques não possam ajustar ou modificar o valor de ID de um objeto para obter acesso a outros dados. O uso de valores de ID de objeto imprevisíveis também pode ajudar, pois eles não são sequenciais e fáceis de adivinhar.

2. Atribuição em massa, exposição de dados e ataques de injeção

Outro tipo de ataque expõe muitos dados em respostas ou permite a modificação de objetos internos por meio de entradas.

A exposição excessiva de dados acontece quando uma API expões amplamente as propriedades do objeto e retorna uma muitos dados em uma resposta, dependendo dos clientes fazerem a solicitação para filtrar dados.

OS invasores podem usar detalhes adicionais da resposta para criar um ataque ainda mais poderoso ou e-mail de phishing. Por exemplo, se uma resposta retorna todos os dados abaixo, ela pode ser usada em e-mails de phishing muito convincentes:

```
{
“Id”: 213,
“FirstName”:”Sanjay”,
“LastName”:”Smythe”,
“EmailAddress”:”ssmythe@hacketyhack.com”,
“Occupation”: “Assistant to the Deputy Associate Vice Sub-undersecretary”,
“DOB”:”1986-05-21”,
“Bank”:”Easygo Financial”,
“AccountNumber”: 1362886306,
“PetName”: “Aloysius”,
 }
```

Ataques de atribuição em massa permitem que chamadas à API alterem ou explorem valores internos quando a API expõe objetos e variáveis internos.

O OWASP explica desta maneira:

> “As estruturas de software às vezes permitem que os desenvolvedores vinculem automaticamente os parâmetros de solicitação HTTP às variáveis ou objetos do código do programa para facilitar o uso dessa estrutura pelos desenvolvedores ... Às vezes, os invasores podem usar essa metodologia para criar novos parâmetros que o desenvolvedor nunca pretendia, o que, por sua vez, cria ou substitui novas variáveis ou objetos no código do programa que não foram planejados”.

O que os desenvolvedores devem fazer? Eles devem entender os possíveis riscos de atribuição em massa durante o desenvolvimento e evitar a exposição de objetos ou variáveis internos que possam ser explorados. As propriedades da lista de permissões, que podem ser atualizadas pelos clientes, também devem ser consideradas.

Aplicativos web já são suscetíveis a ataques de injeção há muito tempo e não é diferente com as APIs. Como os ataques de injeção já são bem conhecidos, não vamos nos alongar, mas basta dizer que entradas precisam ser validadas e higienizadas antes de serem transmitidas. É preciso envidar esforços para usar a prevenção contra vazamento de dados nas respostas da API e limitar o número de registros que podem ser retornados a fim de evitar incidentes de divulgação em massa.

3. Abuso de recursos e APIs invisíveis/nocivas

Outros ataques podem abusar das APIs, consumindo quantidades excessivas de recursos de computação — o que causa sobrecarga e facilita ataques do tipo DoS. A porta fica aberta para esses ataques quando não há limites para aspectos como o número de solicitações por cliente/recurso, registros retornados em uma única resposta ou tamanho da carga da solicitação.

Como vimos nos ataques à JustDial, as APIs de produção podem ser esquecidas e se tornarem invisíveis ou nocivas, pois provavelmente estão desprotegidas e podem ser exploradas. Assim como na segurança em geral, é preciso ter visibilidade do estado da TI ou da superfície de ataque para aplicar os controles de segurança adequados. A visibilidade de todo o nosso estado de endpoints da API não é diferente.

Ao desenvolver APIs, as equipes devem ter um processo refinado para rastrear versões de API para entender quais APIs estão em produção e quais estão obsoletas.

Considerações sobre segurança para APIs

Abordamos o que são as APIs, por que elas são importantes e falamos sobre os ataques que visam as APIs em geral. Agora, vamos ver como a Cloudflare criou segurança para APIs para protegê-las contra esses ataques. Uma segurança para API eficaz precisa ter de tudo, de visibilidade até modelos positivos de segurança para impedir o abuso à proteção de dados.

API Shield da Cloudflare

Fundamentos da visibilidade

Visibilidade

Como com outros aspectos de segurança, temos que ver algo para que possamos protegê-lo. As APIs não são diferentes, principalmente quando as empresas têm centenas ou até milhares de endpoints de API.

A descoberta e a visibilidade das APIs são aspectos básicos fundamentais para gerenciá-las, de modo que as empresas sempre tenham uma visão clara do estado do endpoint da API para evitar que APIs invisíveis ou nocivas se tornem um problema.

Como visto no caso da JustDial, quando empresas não controlam as APIs, podem acontecer violações de dados.

Defesa em profundidade da API

Proteções da camada 7 para APIs

Há muito tempo já implantamos firewalls de aplicativos web para proteger aplicativos contra ataques DDoS na camada 7. A proteção da API deve começar com muitos desses controles dependentes, como limitação de taxa e proteção contra DDoS, a fim de afastar ataques de negação de serviço, tentativas de login por força bruta e o abuso geral realizado por determinados endereços de IP. Dessa forma, você impõe limites de uso à API e garante disponibilidade no combate ao OWASP API 4—Falta de recursos e de limitação de taxa.

As regras de WAF também devem ser usadas para identificar e bloquear ataques comuns às APIs.

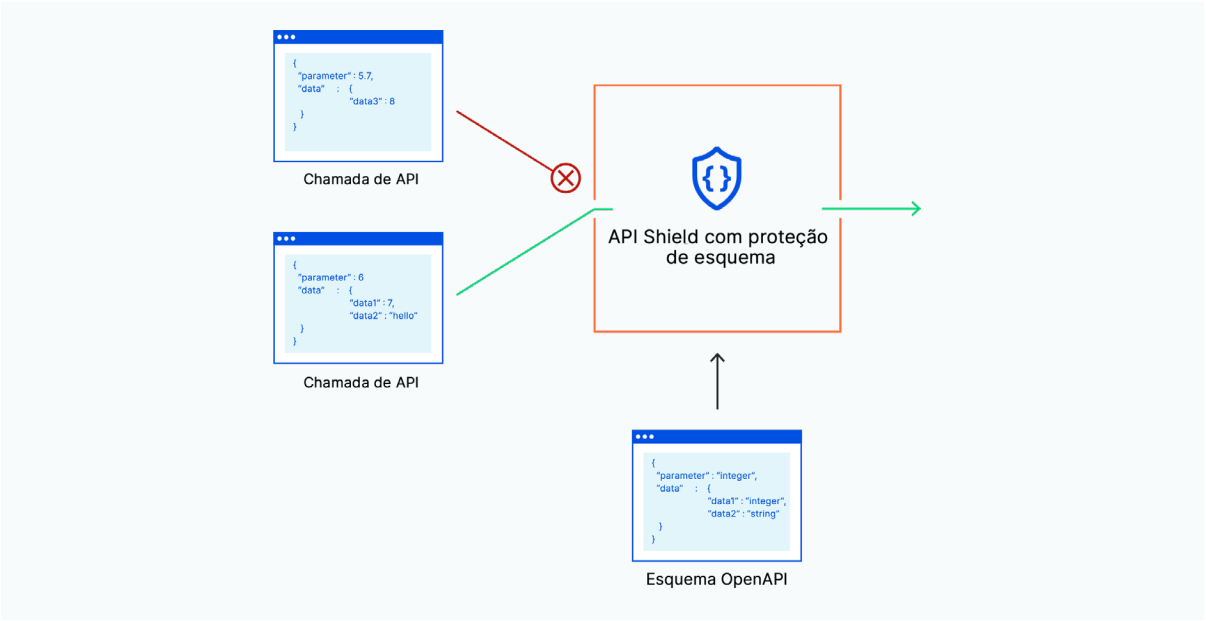
Autenticação e autorização

Autenticação mTLS

Vimos nos ataques à API descritos que a falta de autenticação pode ser devastadora. A autenticação precisa estar integrada desde o início e deve ser reforçada com TLS mútuo para impor identidade baseada em certificados em casos de uso como dispositivos móveis ou de IoT. Essa abordagem é um modelo de lista de permissões mais positivo que permite apenas solicitações de clientes legítimos com certificados válidos.

Verificações de credenciais expostas

As APIs não estão imunes a ataques de preenchimento de credenciais, que variam entre tentativas de login usando credenciais roubadas. Essas credenciais de conta podem ser comprometidas por invasões de terceiros, fora do controle da empresa. Como parte das verificações de autenticação, a segurança da APIs precisa comparar as credenciais no login com um banco de dados de credenciais vazadas. Se elas tiverem algum indício de comprometimento, a segurança da API deve tomar medidas como a redefinição de senha ou a autenticação multifator e, é claro, bloquear a tentativa.



A validação de esquemas avalia cada solicitação em relação ao esquema da API, permitindo ou bloqueando aquelas que não estão de acordo.

Segurança para APIs positiva

Validação de esquema de API

Os desenvolvedores se esforçam muito para criar um esquema de API, que é a documentação ou regras básicas de como eles esperam que os outros interajam com a API. Isso pode estabelecer coisas como métodos de solicitação e operações em cada endpoint (GET /users, POST /users) ou parâmetros de entrada e saída para cada operação. A OpenAPI v3, também conhecida como padrão Swagger, é o esquema mais conhecido de definição de APIs.

Uma segurança de API confiável deve usar um modelo Zero Trust positivo, que imponha o esquema.

Com um esquema em vigor, as solicitações devem ser validadas automaticamente em relação a ele. Todas as operações da API são bloqueadas, exceto aquelas que foram validadas ao estarem em conformidade com o esquema.