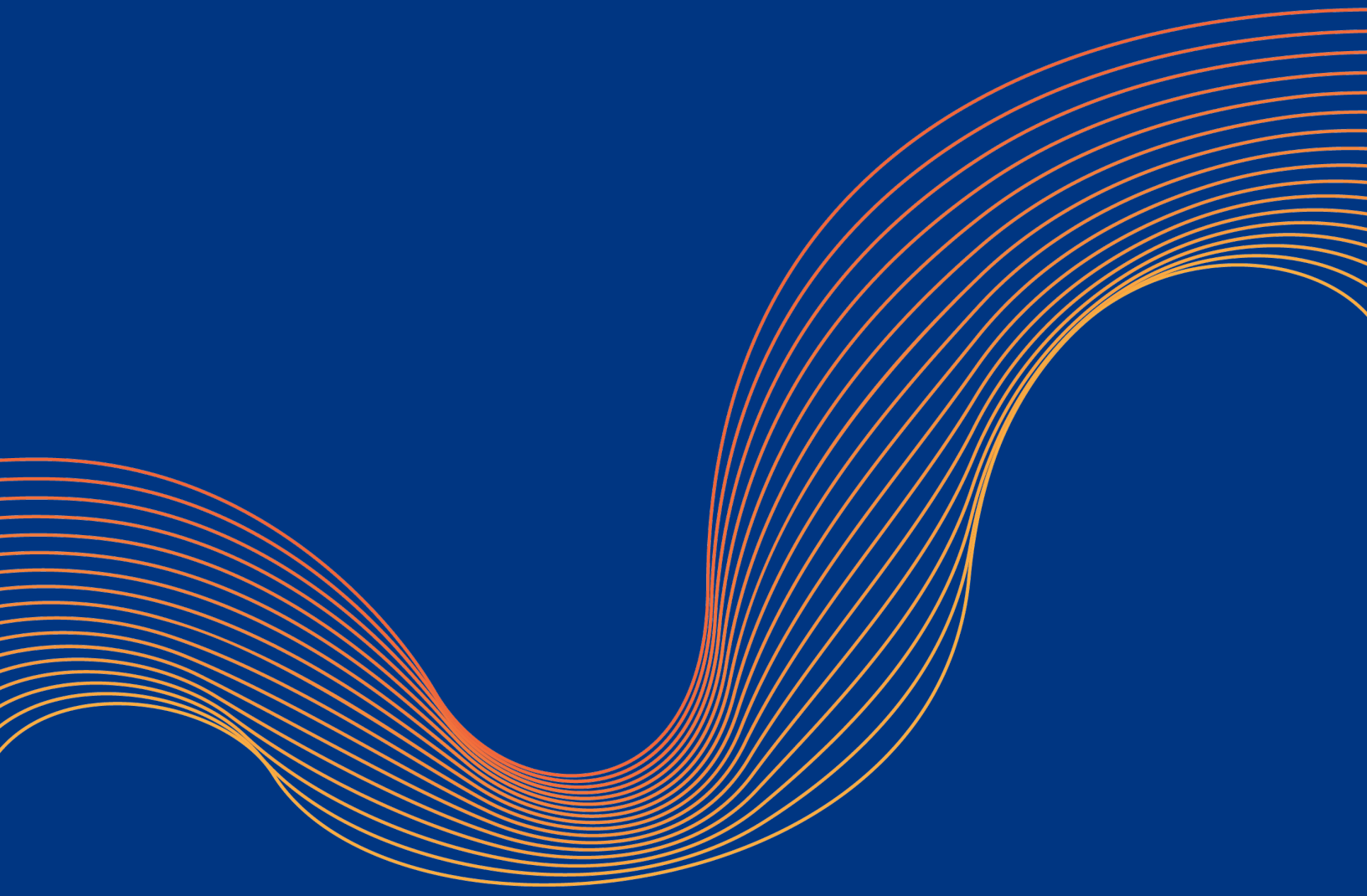
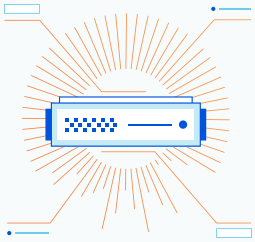
Browser-Isolierung: Häufige Herausforderungen und ihre Überwindung

Die Kombination aus Internet und Zero Trust-Sicherheit



+49 89 2555 2276 | enterprise@cloudflare.com | www.cloudflare.com/de-de/

INHALT



Einleitung

IT-Teams und Sicherheitsexperten misstrauen dem öffentlichen Internet, und das mit gutem Grund: Laut einem Bericht von Verizon gingen 2020 nicht weniger als 39 % aller Datenschutzverletzungen auf Phishing- und Malware-Attacken zurück und einer von Cloudflare bei Forrester Consulting in Auftrag gegebenen Studie kann man entnehmen, dass im gleichen Jahr 61 % aller Unternehmen mit über 1.000 Mitarbeitern mehr Phishing-Angriffe verzeichneten als zuvor.

Jeder, der für Datenverarbeitung und -sicherheit verantwortlich ist, will natürlich dafür sorgen, dass sein Unternehmen nicht in diesen Statistiken auftaucht. Konkret bedeutet das, ...

- **Schadcode und Phishing-Attacken erfolgreich abzuwehren,** auch wenn die Angreifer ihre Taktik ständig ändern, um ihre Aktivitäten zu verschleiern,
- **jeglichen Datenverlust zu unterbinden,** unabhängig davon, ob er durch kompromittierte Geräte oder Benutzerverhalten droht,
- **sich bessere Einblicke in das Surfverhalten der Mitarbeiter zu verschaffen,** um die spezifische Bedrohungslandschaft des Unternehmens besser zu kennen und im Falle von Sicherheitsverletzungen schneller reagieren zu können.

Diese Anwendungsfälle decken sich mit zentralen Zielen des Konzepts der **Zero Trust-Sicherheit**, das Websites und Programmcodes niemals einen Vertrauensvorschuss gewährt und sie stattdessen stets einer abgesicherten Verarbeitung unterwirft, wann immer ein Nutzer mit ihnen interagiert.

Um diese Ziele zu erreichen, setzen einige Unternehmen darauf, das Surfen der Mitarbeiter im Internet durch Browser-Isolierung von den lokalen Netzwerken und der Infrastruktur zu trennen. Dieser Ansatz hat bei lückenloser und effizienter Umsetzung durchaus das Zeug dazu, sich als die bisher wirksamste Methode zur Abwehr von Angriffen aus dem Internet zu erweisen. Doch leider ist er bisher eine Nischentechnologie geblieben, weil er IT-Abteilungen vor so manche Herausforderung stellt, darunter **hohe Kosten, schlechte Nutzererfahrungen, logistische Hürden beim Betrieb** und **Sicherheitslücken.**

Dieses Whitepaper geht näher auf diese Herausforderungen ein und soll IT-Teams und Sicherheitsexperten eine klarere Vorstellung von den Vorkehrungen vermitteln, die sie für einen sicheren Internetzugriff treffen müssen. Außerdem wird eine Methode beschrieben, mit der diese Herausforderungen gemeistert werden können – und zu guter Letzt erfahren Sie, wie wir diese Methode in unser globales Netzwerk integriert haben.

Die Problemfelder gängiger Browser-Isolierungsansätze

In jeder Branche und jedem Sektor müssen sich Unternehmen mit den Themen Malware, Phishing und Datenverlust auseinandersetzen. Browser-Isolierung soll dabei meistens die ersten beiden dieser drei Risiken ausschalten, indem die Blockierlisten, Dateiabgleiche und verhaltensbasierten Ansätze sicherer Web-Gateways ergänzt werden.

In der Praxis allerdings wird dieses Ziel selten erreicht.

Woran liegt das? Um diese Frage zu beantworten, sollte man sich ansehen, wo die gängigen Methoden zur Browser-Isolierung an ihre Grenzen stoßen:

Streaming der Browseroberfläche aus der Cloud

Bei diesem Ansatz, manchmal auch „Pixel-Pushing“ genannt, werden die Vorgänge auf der Browseroberfläche beim Endbenutzer erfasst und an einen in der Cloud gehosteten Remote-Browser übertragen. Durch ihn erfolgt die eigentliche Ausführung der Vorgänge und das Ergebnis wird wiederum in Form von auf Pixeldaten basierenden Bildern des cloudbasierten Browserfensters zurück an den Endbenutzer geschickt. Damit erreicht kein Schadcode das Endgerät – unabhängig davon, ob er manuell oder automatisch heruntergeladen wurde. Außerdem können potenzielle Phishing-Sites mit einer Warnmeldung angezeigt werden, oder ihre Eingabefelder – zum Beispiel für Benutzernamen oder Passwörter – werden blockiert.

Dieser Ansatz eignet sich zwar gut, um Endgeräte von Malware und Phishing abzuschirmen, allerdings bringt er auch die eine oder andere Schwierigkeit mit sich:

Latenz: Erfolgt das Hosting der Remote Browser Isolation in der Public Cloud oder auf einem privaten Netzwerk mit begrenzter geografischer Verteilung, kann bei großen Entfernungen zwischen Endnutzer und Rechenzentren eine hohe Latenz entstehen. Dieses Problem verschärft sich, wenn für den Datenverkehr außerdem ein sicheres Web-Gateway oder andere Sicherheitstools vorgesehen sind, die nicht in denselben Rechenzentren gehostet werden oder aufgrund einer ineffizienten Architektur mehrmalige Containerdurchläufe erfordern.

Hohe Kosten: Das Streaming von Remote-Webseiten für die Endgeräte der Benutzer erfordert eine kontinuierliche Kodierung, die eine sehr große Rechenleistung und damit auch erhebliche Kosten verursacht. Außerdem werden selbst bei umfangreicher Datenoptimierung hohe Bandbreiten benötigt. Diese Kosten werden normalerweise an die Kunden weitergegeben.

Sicherheitslücken: Da das „Pixel-Pushing“ nicht selten die Nutzererfahrung beeinträchtigt, ist sein Einsatz in vielen Firmen nur für Teams verpflichtend, die mit besonders sensiblen Daten arbeiten – zum Beispiel für die Finanz- oder Personalabteilung oder die Unternehmensführung. Denkbar ist auch, dass das Remote-Browsing auf einen kleinen Prozentsatz von Webseiten begrenzt wird, die als besonders gefährlich gelten. In beiden Fällen bleiben Risiken, sei es durch Mitarbeiter, die ohne angemessenen Schutz im Internet unterwegs sind, oder durch vermeintlich vertrauenswürdige Seiten, die kompromittiert wurden.

Notwendigkeit hoher Bandbreiten: Bild-Streaming verlangt eine hohe Bandbreite und kann deshalb die Netzwerkinfrastruktur überlasten oder sich negativ auf die Endnutzererfahrung auswirken. Darüber hinaus geht mit höheren Auflösungen eine exponentielle Steigerung der Pixeldichte einher, sodass insbesondere Textinhalte auf HiDPI-Geräten möglicherweise verschwommen oder unscharf dargestellt werden.

Entfernung von Schadcode durch Aufspaltung von Websites in der Cloud

Dieser Ansatz wird häufig als DOM-Manipulation bezeichnet. Im Bereich der Frontend-Entwicklung bezeichnet das DOM (Document Object Model) die Datenrepräsentation der Struktur- und Inhaltsobjekte einer Webseite. Bei der DOM-Manipulation untersucht ein in der Cloud gehosteter Remote-Browser die Bestandteile einer Webseite (etwa HTML- und CSS-Elemente). Dabei wird versucht, Javascript und anderen aktiven Code, bekannte Exploits sowie sonstige potenziell bösartige Inhalte zu eliminieren. Erst nachdem der Code auf diese Weise bereinigt wurde, wird er an den Browser des Endnutzers zur Darstellung der Seite weitergeleitet. Wie beim „Pixel-Pushing" ist es zudem auch mithilfe der DOM-Manipulation möglich, vor bestimmten Seiten zu warnen, die Phishing-Risiken darstellen.

Da aber bei der DOM-Manipulation nur der Website-Code übertragen und nicht die gesamte Browseroberfläche gestreamt wird, erfordert diese Lösung weniger Bandbreite, sodass aufgrund kürzerer Ladezeiten eine bessere Endnutzererfahrung geboten werden kann.

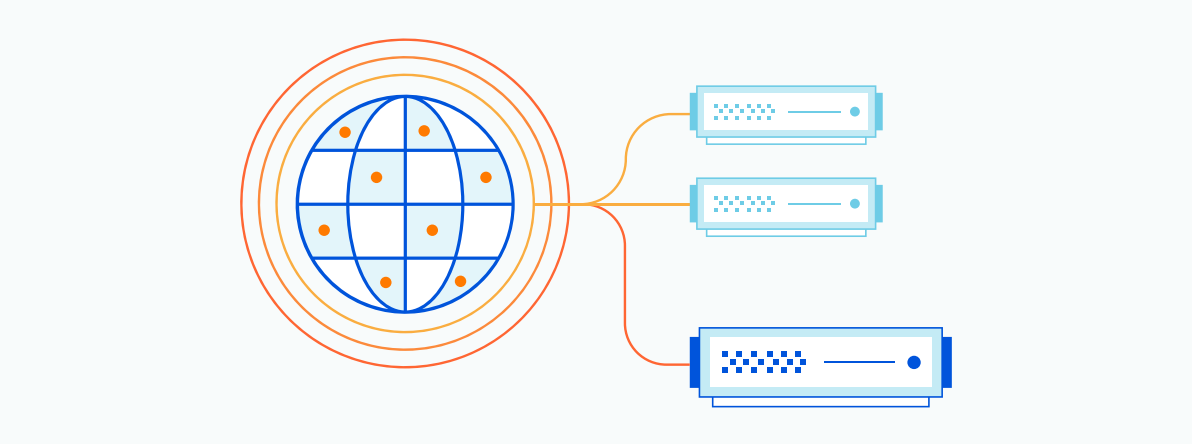
Trotzdem hat auch dieser Ansatz seine Tücken:

Latenz: Wie beim „Pixel-Pushing" kann eine hohe Latenz immer noch dadurch entstehen, dass die Browser-Isolierung mit DOM-Manipulation auf einer Public Cloud oder einem privaten Netzwerk mit begrenzter geografischer Verteilung basiert und die Entfernung zu den Ursprungsservern zu groß ist oder die Browser-Isolierung und andere Sicherheitstools in verschiedenen Rechenzentren gehostet werden.

Nicht funktionierende Websites: Der Versuch, bösartigen aktiven Code zu entfernen – und dann HTML und CSS zu rekonstruieren sowie auch ungewöhnliche Site-Architekturen neu aufzubauen – beeinträchtigt die Funktionsfähigkeit von Webseiten, die dann nicht richtig dargestellt oder gar nicht mehr gerendert werden. Und selbst wenn diese Probleme ausbleiben, nimmt der Website-Betreiber möglicherweise regelmäßig Änderungen vor, sodass die heute noch erfolgreiche DOM-Rekonstruktion schon morgen nicht mehr funktioniert. DOM-Manipulationen lassen sich sogar mit gängigen und unternehmensweit eingesetzten Diensten wie Google G Suite oder Microsoft Office 365 nicht ohne Weiteres realisieren. Im Ergebnis müssen IT-Abteilungen immer wieder neue Folgeprobleme lösen – eine Sisyphusarbeit, die erhebliche IT-Ressourcen beansprucht.

Hohe Kosten: Einige DOM-Manipulationsdienste werden in der Public Cloud-Infrastruktur von Drittanbietern gehostet, was zusätzliche Kosten verursacht, die in der Regel an die Kunden weitergegeben werden, obwohl diese nach wie vor Risiken ausgesetzt sind – sei es durch Mitarbeiter, die ohne angemessenen Schutz im Internet unterwegs sind, oder durch vermeintlich vertrauenswürdige Seiten, die kompromittiert wurden.

Sicherheitslücken: Wie beim „Pixel-Pushing" sorgt auch bei der DOM-Manipulation die schwankende Qualität der Endnutzererfahrung dafür, dass Unternehmen diese Lösung oft nur sporadisch verwenden. Obwohl DOM-Manipulation als eine Form der Browser-Isolierung gilt, wird dabei nicht vertrauenswürdiger Fremdcode an Endgeräte gesendet. Wenn der Dienst bösartigen Quelltext übersieht – ein permanentes Risiko angesichts der sich ständig verändernden Bedrohungslandschaft –, können Endgeräte trotzdem infiziert werden.



Isolierter Browserbetrieb mit einer virtuellen Maschine auf dem Endgerät

Bei diesem Ansatz erstellt eine auf dem Endgerät installierte Software eine virtuelle Maschine, die vom restlichen Betriebssystem des Geräts isoliert ist. Alle Browser-Vorgänge finden auf dieser virtuellen Maschine statt, sodass heruntergeladene Malware andere Bereiche des Geräts nicht infizieren kann. Mit den weiteren vorgestellten Ansätzen hat diese Lösung gemeinsam, dass die Software auch bestimmte Seiten als Phishing-Risiko kennzeichnen kann.

Doch leider bringt auch dieser Ansatz Probleme mit sich:

Hohe Anforderungen an CPU und RAM: Durch den Betrieb einer separaten virtuellen Maschine werden viele PCs ausgebremst, sodass das Surfen für den Endnutzer zur Geduldsprobe werden kann.

Probleme mit der Endpunktverwaltung: Browser-Isolierung per lokaler Software setzt voraus, dass IT-Teams diese Software auf jedem Endpunktgerät installieren und aktuell halten – keine leichte Aufgabe insbesondere bei hohen Mitarbeiterzahlen. Noch komplexer wird es, wenn das Unternehmen viele Remote-Mitarbeiter beschäftigt oder mit Auftragnehmern zusammenarbeitet, die ihre eigenen Geräte verwenden.

Kompatibilitätsprobleme mit Mobilgeräten: Für eine Browser-Isolierung mithilfe einer lokal installierten virtuellen Maschine sind betriebssystemspezifische Implementierungen nötig. Mobile Geräte werden dabei oft nicht unterstützt.

Durchbrochene Isolierung: Bei lokalen Browser-Isolierungsdiensten treten regelmäßig Sicherheitslücken auf, die es bösartigem Code ermöglichen, auf das Hauptbetriebssystem zuzugreifen. Die IT-Abteilung oder Endnutzer müssen dann Patches und Updates manuell installieren. Kommt es dabei zu Fehlern, ist das Endgerät möglicherweise weiterhin anfällig für Angriffe.

Schlechte Endnutzererfahrung: Bei auf virtuellen Maschinen basierenden Implementierungen müssen die Endnutzer oft separate „virtualisierte“ Browser, Fenster oder Desktops verwenden. Die Folge sind ein erhöhter Schulungsbedarf und mehr Anfragen beim IT-Support.

Längerfristige Folgeprobleme

Zu den unmittelbar auftretenden Problemen mit den gängigen Ansätzen zur Browser-Isolierung – von den hohen Kosten über die schlechten Endnutzererfahrungen bis hin zu den Schwierigkeiten mit der Endgeräteverwaltung – gesellen sich weitere Herausforderungen.

Längerfristige Risiken ergeben sich zum Beispiel aus dem Einsatz von unsicheren Endgeräten. Wenn sich ein Endnutzer mit einem kompromittierten Gerät bei sensiblen Anwendungen anmeldet, kann Malware vom Anwender unbemerkt auf die entsprechenden Daten zugreifen.

Bestimmte Arten von Datenverlusten lassen sich zudem mit den bisher vorgestellten Ansätzen nur schwer verhindern. So kann jeder interne Nutzer vertrauliche Informationen nach wie vor hochladen und per E-Mail versenden oder in Online-Formulare eingeben.

Eine bessere Lösung für die Remote Browser Isolation

Trotz der im vorherigen Abschnitt beschriebenen Probleme wollen Unternehmen das Remote-Browsing nutzen.

Glücklicherweise gibt es bestimmte Technologien, um die genannten Herausforderungen anzugehen:

Problem	Lösung
Latenz	**Verwendung eines großen Edge-Netzwerks:** Anstatt die Browser-Isolierung in einer Public Cloud mit nur wenigen Rechenzentren zu hosten, empfiehlt es sich, auf ein globales Edge-Netzwerk mit stets geringen Abständen zu Ihren Endnutzern zu setzen. Sorgen Sie außerdem dafür, dass Ihre Software für Browser-Isolierung in denselben Rechenzentren betrieben wird wie Ihre sicheren Web-Gateways und andere Sicherheitstools.
Notwendigkeit hoher Bandbreiten	**Kein Pixel-Pushing:** Das Streamen der Oberfläche eines Remote-Browsers ist für größere Unternehmen sowohl aus Kostengründen als auch aufgrund der schlechten Endnutzererfahrung keine zweckmäßige Lösung.
Nicht funktionierende Websites	**Einsatz nativer Browsertechnologie:** Remote-Browser, die dieselbe Technologie verwenden wie gängige Applikationen auf den Endgeräten, sind bei der korrekten Rekonstruktion der verschiedensten Arten von Websites zuverlässiger.
Hohe Rechenleistungskosten	**Cloud Computing der nächsten Generation:** Vermeiden Sie das Hosting einer Remote Browser Isolation in einer Public Cloud. Setzen Sie außerdem auf effiziente Serverless Computing-Ansätze, um die Virtualisierung und Containerisierung zu optimieren, denn dann können Sie die zugrundeliegenden Serverressourcen wirksamer einsetzen, weil deren Abstimmung und Verwaltung wegfällt.
Sicherheitslücken	**Einsatz nativer Browsertechnologie:** Anstatt sich mit der Entscheidung aufzuhalten, welcher Code übertragen werden soll und welcher nicht, kann native Browsertechnologie das Senden von Quelltext komplett vermeiden und sich auf den letzten Schritt im Rendering-Prozess beschränken, der die Seite darstellt.
Endpunkt-spezifische Schwierigkeiten	**Keine lokale Browser-Isolierung:** Die Browser-Isolierung auf Endgeräten ist zu langsam, schwierig zu verwalten und ein vollkommen veralteter Ansatz.

Wenn Sie sehen möchten, wie diese Technologien in der Praxis funktionieren, empfehlen wir Ihnen zum Beispiel Cloudflare Browser Isolation, eine in unsere Zero Trust-Plattform nativ eingebundene Lösung.

Der Ansatz von Cloudflare für kostengünstiges und anwenderfreundlicheres Remote-Browsing

Allgemein vermeidet Cloudflare endpunktspezifische Schwierigkeiten schon allein dadurch, dass unsere Lösung in einem globalen Edge-Netzwerk und nicht auf den Endgeräten betrieben wird. Im Detail zeigt sich außerdem, dass die Technologie von Cloudflare auch die anderen Probleme beseitigt:

Problem	Lösung	Umsetzung durch Cloudflare
 Latenz	Verwendung eines großen Edge-Netzwerks	Unsere Remote Browser Isolation steht in jedem Rechenzentrum des Edge-Netzwerks von Cloudflare zur Verfügung und damit an über 200 Standorten in 100 Ländern. 95 % der weltweit mit dem Internet verbundenen Menschen erreichen unser Netzwerk innerhalb von 100 Millisekunden. Über dieselbe Infrastruktur stellen wir auch globale DNS- und CDN-Dienste mit extrem geringer Latenz bereit. Darüber hinaus greifen unser Remote-Browsing und unser Forward-Proxy nahtlos ineinander, um alle anderen Filter (zum Beispiel das Blockieren eines Teils einer Seite) und Prüfungen anzuwenden, ohne dass dafür mehrere Durchläufe und Hops zwischen Lösungen an verschiedenen Punkten erforderlich sind.
 Notwendigkeit hoher Bandbreiten	Native Browser-Technologie	Network Vector Rendering (NVR; siehe unten) ist eine Technologie von Cloudflare, die nicht Pixeldaten oder durch Scrubbing-Prozesse bereinigten Quelltext weiterleitet, sondern Darstellungsbefehle. Diese Methode erfordert nur einen Bruchteil der bei normalem Browsing, DOM-Manipulationen oder gar „Pixel-Pushing“ erforderlichen Bandbreite.
 Nicht funktionierende Websites	Native Browser-Technologie	Der Remote-Browser von Cloudflare beruht (wie etwa auch Google Chrome und 21 weitere gängige Webbrowser) auf der Open-Source-Engine Chromium. In die Weiterentwicklung von Chromium wird nach wie vor viel investiert, sodass für ein Höchstmaß an Website-Kompatibilität gesorgt ist. Da Cloudflare mit Network Vector Rendering eine Technologie einsetzt, die Darstellungsbefehle statt „bereinigtem“ Code überträgt, ist außerdem gewährleistet, dass selbst die komplexesten Webseiten korrekt angezeigt werden.
 Hohe Rechenleistungskosten	Cloud Computing der nächsten Generation	Da die Cloudflare Browser Isolation in unserem eigenen Netzwerk betrieben wird, müssen wir keine zusätzlichen Kosten für die Nutzung einer Public Cloud an unsere Kunden weitergeben. Weil wir zudem unsere Serverressourcen sehr effizient aufeinander abstimmen und verwalten können, gelingt es uns, die Kaltstarts zu vermeiden, die bei in einer Public Cloud gehosteten Anwendungen häufig mehrere Sekunden in Anspruch nehmen.
 Sicherheitslücken	Native Browser-Technologie	Cloudflare überträgt keinen eigenständigen Website-Code, sondern vektorbasierte sparsame Darstellungsbefehle. Das Risiko der Ausführung von nicht vertrauenswürdigem Code auf dem Endgerät fällt damit weg. Unerkannte Malware kompromittiert allenfalls den Remote-Browser, nicht jedoch den Endpunkt. Und da Cloudflare hervorragende Endnutzererfahrungen bietet, können Unternehmen das Remote-Browsing auch auf Anwendungsfälle ausweiten, die bisher wegen des geringeren Risikos außen vor geblieben sind und deshalb möglicherweise keinen ausreichenden Schutz genossen haben.
	Detaillierte Kontrolle des Nutzerverhaltens	Tools, die Datenverluste verhindern sollen, schützen für gewöhnlich die Datenübertragung im Netzwerk, indem sie sie entweder zulassen oder blockieren. Mit Cloudflare Browser Isolation können Administratoren folgende Einstellungen und Vorgänge im Detail kontrollieren: • Berechtigungen zum Kopieren/Einfügen/Drucken • Uploads/Downloads • allgemeine Tastatureingaben • Berechtigungen für Formulareingaben • Speicherort für Downloads* * demnächst erhältlich

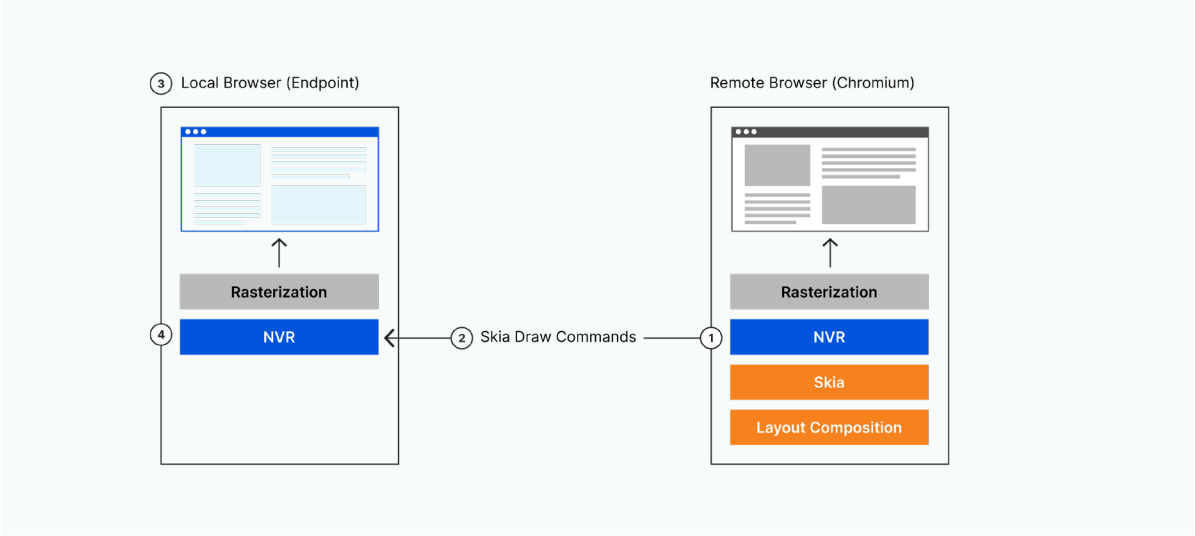
Lesen Sie weiter, um tiefer in die oben beschriebene NVR-Technik einzusteigen:

Network Vector Rendering verbessert die Nutzererfahrung und schließt Sicherheitslücken

Wie bereits erwähnt, basiert der Remote-Browser von Cloudflare auf Chromium. Eine wichtige architektonische Eigenschaft des Chromium-Browsers ist der Einsatz von Skia – einer häufig und plattformübergreifend verwendeten Grafik-Engine für Android, Google Chrome, Chrome OS, Mozilla Firefox und viele andere Browser. Alle HTML5-kompatiblen Browser können Skia rendern und alles, was in einem Browserfenster von Chromium zu sehen ist, wird über die Rendering-Ebene von Skia dargestellt – und zwar nicht nur die Benutzeroberfläche der Anwendung (wie zum Beispiel die Menüs), sondern insbesondere auch der gesamte Inhalt des Webseitenfensters. Cloudflare kann heruntergeladene Dateien sogar isolieren und entsprechend den Anforderungen des Endnutzers an verschiedene Speicherorte verschieben.

Mit Network Vector Rendering (NVR) von Cloudflare werden die Skia-Darstellungsbefehle des in der Cloud betriebenen Chromium-Browsers abgefangen, tokenisiert und komprimiert. Dann werden sie verschlüsselt an den HTML5-kompatiblen Webbrowser übertragen, der auf dem Desktopcomputer oder Mobilgerät des Nutzers am Endpunkt betrieben wird. Bei den von NVR erfassten Skia-API-Anweisungen handelt es sich um vorgerasterte und damit sehr kompakte Befehle. Da Skia so weit verbreitet ist, funktioniert das Remote-Browsing von Cloudflare mit jedem modernen Webbrowser.

Darüber hinaus bietet Network Vector Rendering Sicherheitsvorteile. Wie erwähnt liefert Cloudflare Darstellungsbefehle und nicht den eigentlichen Quelltext der Website an die Endgeräte. Deshalb bietet der dem Prozess zugrundeliegende Datentransport keine Angriffsfläche.



Weitere Informationen und nächste Schritte

Wir von Cloudflare sind angetreten, um ein besseres Internet zu schaffen und dafür zu sorgen, dass Technologien, die bisher nur großen Unternehmen mit hochentwickelten Netzwerken, spezialisierten IT-Teams und riesigen Budgets vorbehalten waren, breiteren Einsatz finden. Die Bereitstellung einer besseren Lösung für die Remote Browser Isolation ist ein wichtiger Bestandteil dieser Demokratisierungsmission.

Indem wir die Kosten für die Browser-Isolierung senken – und gleichzeitig außergewöhnlich gute Endnutzererfahrungen bieten – hoffen wir, dass der wahre Wert dieser Technologie einer steigenden Zahl von Unternehmen zugänglich wird. Angesichts des Paradigmenwechsels hin zum Zero Trust-Browsing sind wir davon überzeugt, dass blindes Vertrauen in willkürlichen Website-Code bald genauso archaisch anmuten wird wie die (gar nicht so weit zurückliegenden) Zeiten, in denen die HTTPS-Verschlüsselung „sensiblen“ Login-Seiten und E-Commerce-Checkouts vorbehalten war.

Sie finden unter folgendem Link weitere Angaben zu Cloudflare Browser Isolation und können sich informieren, wie mit dieser Lösung Zero Trust-Browsing verwirklicht werden kann: https://www.cloudflare.com/products/zero-trust/browser-isolation/