

Verstehen des Paketflusses in einer sicheren Web-Appliance

Inhalt

[Einleitung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Abgrenzungstypen der Proxybereitstellung](#)

[TLS-Handshake](#)

[HTTP-Antwortcode](#)

[1xx : Informativ](#)

[2xx: Erfolgreich](#)

[3xx:Umleitung](#)

[4xx-Codes: Client-Fehler](#)

[5xx: Serverfehler](#)

[Explizite Bereitstellung](#)

[HTTP-Datenverkehr bei expliziter Bereitstellung ohne Authentifizierung](#)

[Client und SWA](#)

[SWA und Webserver](#)

[Datenverkehr mit zwischengespeicherten Daten](#)

[HTTP-Datenverkehr bei expliziter Bereitstellung ohne Authentifizierung](#)

[Client und SWA](#)

[SWA und Webserver](#)

[HTTPS-Passthrough-Datenverkehr](#)

[Transparente Bereitstellung](#)

[HTTP-Datenverkehr bei transparenter Bereitstellung ohne Authentifizierung](#)

[Client und SWA](#)

[SWA und Webserver](#)

[Datenverkehr mit zwischengespeicherten Daten](#)

[HTTP-Datenverkehr bei transparenter Bereitstellung ohne Authentifizierung](#)

[Client und SWA](#)

[SWA und Webserver](#)

[Zugehörige Informationen](#)

Einleitung

In diesem Dokument wird der Netzwerkfluss in einem Proxy-konfigurierten Netzwerk beschrieben. Der Schwerpunkt liegt dabei auf der Secure Web Appliance (SWA).

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Grundlegende TCP/IP-Konzepte
- Grundkenntnisse der Proxy-Einrichtung.
- Grundkenntnisse des Authentifizierungsmechanismus, der in der Umgebung mit Proxy verwendet wird.

Verwendete Abkürzungen sind diese Artikel:

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

IP: Internetprotokoll

GRE: Generic Routing Encapsulation

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

URL: Uniform Resource Locator

TLS: Transport Layer Security

Verwendete Komponenten

Dieses Dokument ist nicht auf bestimmte Software- und Hardware-Versionen beschränkt.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Abgrenzungstypen der Proxybereitstellung

TLS-Handshake

Ein TLS-Handshake in HTTPS tritt auf, wenn ein Client und ein Server über das Internet kommunizieren und so eine sichere Verbindung herstellen. Der Prozess schützt den Datenschutz und die Datenintegrität zwischen zwei kommunizierenden Anwendungen. Es arbeitet mit einer Reihe von Schritten, in denen Client und Server sich auf Verschlüsselungsstandards und -codes für alle nachfolgenden Übertragungen einigen. Der Handshake soll vor unbefugtem Zugriff oder Manipulation durch Dritte abschrecken. Es authentifiziert auch die Identität der kommunizierenden Parteien, um Identitätswechsel zu beseitigen. Dieser Prozess ist für HTTPS von entscheidender Bedeutung, da er die Sicherheit der Daten bei der Übertragung gewährleistet.

Dies sind die Schritte eines TLS-Handshakes:

1. Client Hello: Der Client initiiert den Handshake-Prozess mit einer Hello-Nachricht. Diese Nachricht enthält die Client-TLS-Version, unterstützte Verschlüsselungssuiten und eine zufällige Bytefolge, die als "client random" bezeichnet wird.
2. Server Hello: Der Server antwortet mit einer Hello-Nachricht. Diese Nachricht enthält die vom Server ausgewählte TLS-Version, die ausgewählte Verschlüsselungs-Suite, eine als "Server random" bekannte Zeichenfolge mit zufälligen Bytes und das digitale Serverzertifikat. Bei Bedarf fordert der Server auch das digitale Client-Zertifikat für die gegenseitige Authentifizierung an.
3. Client verifiziert das Serverzertifikat: Der Client überprüft das digitale Serverzertifikat bei der Zertifizierungsstelle, die es ausgestellt hat. Dadurch wird sichergestellt, dass der Client mit dem legitimen Server kommuniziert.
4. Pre-Master Secret: Der Client sendet eine zufällige Byte-Zeichenfolge, die als "Pre-Master Secret" bezeichnet wird und zur Erstellung der Sitzungsschlüssel beiträgt. Der Client verschlüsselt diesen Pre-Master-Schlüssel mit dem öffentlichen Schlüssel des Servers, sodass nur der Server ihn mit seinem privaten Schlüssel entschlüsseln kann.
5. Master Secret: Sowohl der Client als auch der Server verwenden den Pre-Master Secret und die zufälligen Byte-Strings aus den Hello-Nachrichten, um unabhängig voneinander den gleichen "Master Secret" zu berechnen. Dieser gemeinsame geheime Schlüssel ist die Grundlage für die Generierung der Sitzungsschlüssel.
6. Client Finished: Der Client sendet eine mit dem Sitzungsschlüssel verschlüsselte "Finished"-Nachricht, um den Abschluss des Client-Teils des Handshakes zu signalisieren.
7. Server Finished (Server beendet): Der Server sendet eine "Finished"-Nachricht, die ebenfalls mit dem Sitzungsschlüssel verschlüsselt ist, um den Abschluss des Server-Teils des Handshakes zu signalisieren.

HTTP-Antwortcode

1xx : Informativ

Code	Details
100 Fortfahren	Typisch für das ICAP-Protokoll. Dies ist eine Informationsantwort, die den Client darüber informiert, dass er weiterhin Daten senden kann. Bei ICAP-Diensten (wie z. B. Virenskans) kann der Server nur die erste x Byte menge sehen wollen. Wenn das Scannen der ersten Bytes abgeschlossen ist und kein Virus entdeckt wurde, sendet es den Befehl 100 Continue, um den Client zu informieren und den Rest des Objekts zu senden.

2xx: Erfolgreich

Code	Details
200 OK	Der gängigste Antwortcode. Dies bedeutet, dass die Anforderung problemlos erfolgreich ist.

3xx: Umleitung

Code	Details
301 Permanente Weiterleitung	Dies ist eine permanente Umleitung. Sie können diesen Code sehen, wenn Sie zu www sub-domain umleiten.
302 Temporäre Umleitung	Dies ist eine vorübergehende Umleitung. Der Client wird angewiesen, eine neue Anforderung für das im Location:-Header angegebene Objekt zu erstellen.
304 Nicht geändert	Dies ist als Antwort auf ein GIMS (GET If-modified-since). Hierbei handelt es sich buchstäblich um eine Standard-HTTP-GET, die den Header If-modified-since: <date> enthält. Dieser Header teilt dem Server mit, dass sich im lokalen Cache des Clients eine Kopie des angeforderten Objekts befindet, die das Datum enthält, an dem das Objekt abgerufen wurde. Wenn das Objekt seit diesem Datum geändert wurde, antwortet der Server mit 200 OK und einer neuen Kopie des Objekts. Wenn sich das Objekt seit dem Abrufdatum nicht geändert hat, sendet der Server eine 304 Not Modified-Antwort zurück.
307-Authentifizierungsumleitung	Dies zeigt sich vor allem bei der transparenten Proxy-Bereitstellung, wenn der Proxy-Server so konfiguriert ist, dass er die Anforderung authentifiziert und die Anforderung an eine andere URL umleitet, um den Benutzer zu authentifizieren.

4xx-Codes: Client-Fehler

Code	Details
400 Ungültige Anforderung	Dies deutet auf ein Problem mit der HTTP-Anfrage hin, da sie nicht der richtigen Syntax entspricht. Mögliche Gründe können mehrere Header in einer einzelnen Zeile, Leerzeichen innerhalb eines Headers oder das Fehlen von HTTP/1.1 im URI sein. Die richtige

	Syntax finden Sie in RFC 2616.
401 Nicht autorisiert Webserverauthentifizierung erforderlich	<p>Der Zugriff auf das angeforderte Objekt erfordert eine Authentifizierung. Der 401-Code wird zur Authentifizierung mit einem Ziel-Webserver verwendet. Wenn der SWA im transparenten Modus arbeitet und die Authentifizierung auf dem Proxy aktiviert ist, gibt er eine 401 an den Client zurück, da sich die Appliance so darstellt, als wäre sie der OCS (Original Content Server).</p> <p>Die Authentifizierungsmethoden, die verwendet werden können, werden in einem HTTP-Antwort-Header "www-Authenticate:" beschrieben. Dadurch wird der Client informiert, ob der Server NTLM, eine einfache oder eine andere Form der Authentifizierung anfordert.</p>
403 abgelehnt	Der Client kann nicht auf das angeforderte Objekt zugreifen. Verschiedene Gründe können dazu führen, dass ein Server den Objektzugriff verweigert. Der Server liefert in der Regel eine Ursachenbeschreibung innerhalb der HTTP-Daten- oder HTML-Antwort.
404 Nicht gefunden	Das angeforderte Objekt ist auf dem Server nicht vorhanden.
407 Proxy-Authentifizierung erforderlich	<p>Dies ist dasselbe wie ein 401, mit der Ausnahme, dass es speziell für die Authentifizierung an einem Proxy und nicht dem OCS dient. Diese wird nur gesendet, wenn die Anforderung explizit an den Proxy gesendet wurde.</p> <p>Ein 407 kann nicht an einen Client gesendet werden, während SWA als transparenter Proxy konfiguriert ist, da der Client nicht weiß, dass der Proxy vorhanden ist. Wenn dies der Fall ist, wird der Client höchstwahrscheinlich den TCP-Socket FIN oder RST starten.</p>

5xx: Serverfehler

Code	Details
501 Interner Serverfehler	Generischer Webserverfehler.
502 Ungültiges Gateway	Tritt auf, wenn ein Server, der als Gateway oder Proxy fungiert, eine ungültige Antwort von einem eingehenden Server empfängt. Es signalisiert, dass das Gateway eine unangemessene Antwort

	vom Upstream- oder Ursprungsserver erhalten hat.
503 Dienst nicht verfügbar	Zeigt an, dass der Server derzeit aufgrund einer vorübergehenden Überlastung oder geplanter Wartung nicht in der Lage ist, die Anforderung zu bearbeiten. Dies bedeutet, dass der Server vorübergehend außer Betrieb ist, aber nach einiger Zeit wieder verfügbar sein kann.
504 Gateway-Zeitüberschreitung	Zeigt an, dass ein Client oder Proxy keine zeitnahe Antwort vom Webserver erhalten hat, auf den er zugreifen wollte, um die Webseite zu laden oder eine andere Anforderung des Browsers zu erfüllen. Dies bedeutet oft, dass der Upstream-Server ausgefallen ist.

Explizite Bereitstellung

Hier

HTTP-Datenverkehr bei expliziter Bereitstellung ohne Authentifizierung

Client und SWA

Der Netzwerkverkehr verläuft zwischen der IP-Adresse des Clients und der IP-Adresse der SWA-Proxyschnittstelle (in der Regel handelt es sich um die P1-Schnittstelle, kann aber auch die P2- oder die Management-Schnittstelle sein, hängt von der Proxy-Konfiguration ab).

Der Datenverkehr vom Client ist für TCP-Port 80 oder 3128 zum SWA bestimmt (Standard-SWA-Proxyports sind TCP 80 und 3128, in diesem Beispiel verwenden wir Port 3128).

- TCP-Handshake
- HTTP Get from Client (Ziel-IP = SWA-IP, Ziel-Port = 3128)
- HTTP-Antwort vom Proxy (Quell-IP = SWA)
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
12544	2024-01-25 09:35:25.989719	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	78	2	65238 → 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=1762371780 TSecr=0 SACK_PERM
12545	2024-01-25 09:35:25.989748	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	74	2	3128 → 65238 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1360 WS=64 SACK_PERM TSval=322700886 TSecr=1762371849
12567	2024-01-25 09:35:26.046546	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=1 Ack=1 Win=132288 Len=0 TSval=1762371848 TSecr=3227008837
12568	2024-01-25 09:35:26.046877	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	HTTP	188	2	GET http://example.com/ HTTP/1.1
12569	2024-01-25 09:35:26.046945	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [ACK] Seq=1 Ack=123 Win=65408 Len=0 TSval=3227008847 TSecr=1762371849
12851	2024-01-25 09:35:26.286288	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	1254	2	3128 → 65238 [ACK] Seq=1 Ack=123 Win=65408 Len=1188 TSval=3227001886 TSecr=1762371849 [TCP
12852	2024-01-25 09:35:26.286297	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	HTTP	599	2	HTTP/1.1 200 OK (text/html)
12992	2024-01-25 09:35:26.347713	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=123 Ack=1189 Win=131072 Len=0 TSval=1762372145 TSecr=3227001886
12993	2024-01-25 09:35:26.347815	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=123 Ack=1722 Win=130560 Len=0 TSval=1762372145 TSecr=3227001886
12994	2024-01-25 09:35:26.353174	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	66	2	65238 → 3128 [FIN, ACK] Seq=123 Ack=1722 Win=131072 Len=0 TSval=1762372150 TSecr=3227001886
12995	2024-01-25 09:35:26.353217	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [ACK] Seq=1722 Ack=124 Win=65408 Len=0 TSval=3227001147 TSecr=1762372150
12996	2024-01-25 09:35:26.353397	10.48.48.185	Vmware_Bd:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [FIN, ACK] Seq=1722 Ack=124 Win=65408 Len=0 TSval=3227001147 TSecr=1762372150
12997	2024-01-25 09:35:26.412438	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_Bd:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=124 Ack=1723 Win=131072 Len=0 TSval=1762372212 TSecr=3227001147

Image-Client an SWA, expliziter HTTP-Modus

SWA und Webserver

Der Netzwerkverkehr erfolgt zwischen der IP-Adresse des Proxys und der IP-Adresse des Webserver.

Der Datenverkehr von SWA wird an TCP-Port 80 weitergeleitet und von einem zufälligen Port (nicht dem Proxy-Port) empfangen.

- TCP-Handshake
- HTTP Get from Proxy (Ziel-IP = Webserver, Ziel-Port = 80)
- HTTP-Antwort vom Webserver (Quell-IP = Proxyserver)
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
12570	2024-01-25 09:35:26.053195	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	74	3	23146 → 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1360 WS=64 SACK_PERM TSval=3190021713 TSecr=0
12778	2024-01-25 09:35:26.168035	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	74	3	80 → 23146 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1360 SACK_PERM TSval=2163592063 TSecr=172700883
12779	2024-01-25 09:35:26.168077	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=1 Ack=1 Win=13568 Len=0 TSval=3190021832 TSecr=2163592063
12780	2024-01-25 09:35:26.168172	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	HTTP	242	3	GET / HTTP/1.1
12833	2024-01-25 09:35:26.280446	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	3	80 → 23146 [ACK] Seq=1 Ack=177 Win=67072 Len=0 TSval=2163592176 TSecr=3190021832
12834	2024-01-25 09:35:26.281757	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	1414	3	80 → 23146 [ACK] Seq=1 Ack=177 Win=67072 Len=1348 TSval=2163592177 TSecr=3190021832 [TCP seq
12835	2024-01-25 09:35:26.281789	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=177 Ack=1349 Win=12224 Len=0 TSval=3190021942 TSecr=2163592177
12836	2024-01-25 09:35:26.281793	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	HTTP	325	3	HTTP/1.1 200 OK (text/html)
12837	2024-01-25 09:35:26.281801	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=177 Ack=1608 Win=11968 Len=0 TSval=3190021942 TSecr=2163592177

Image - HTTP-SWA zu Webserver-Explicit-kein Cache

Hier ist ein Beispiel für HTTP Get from Client

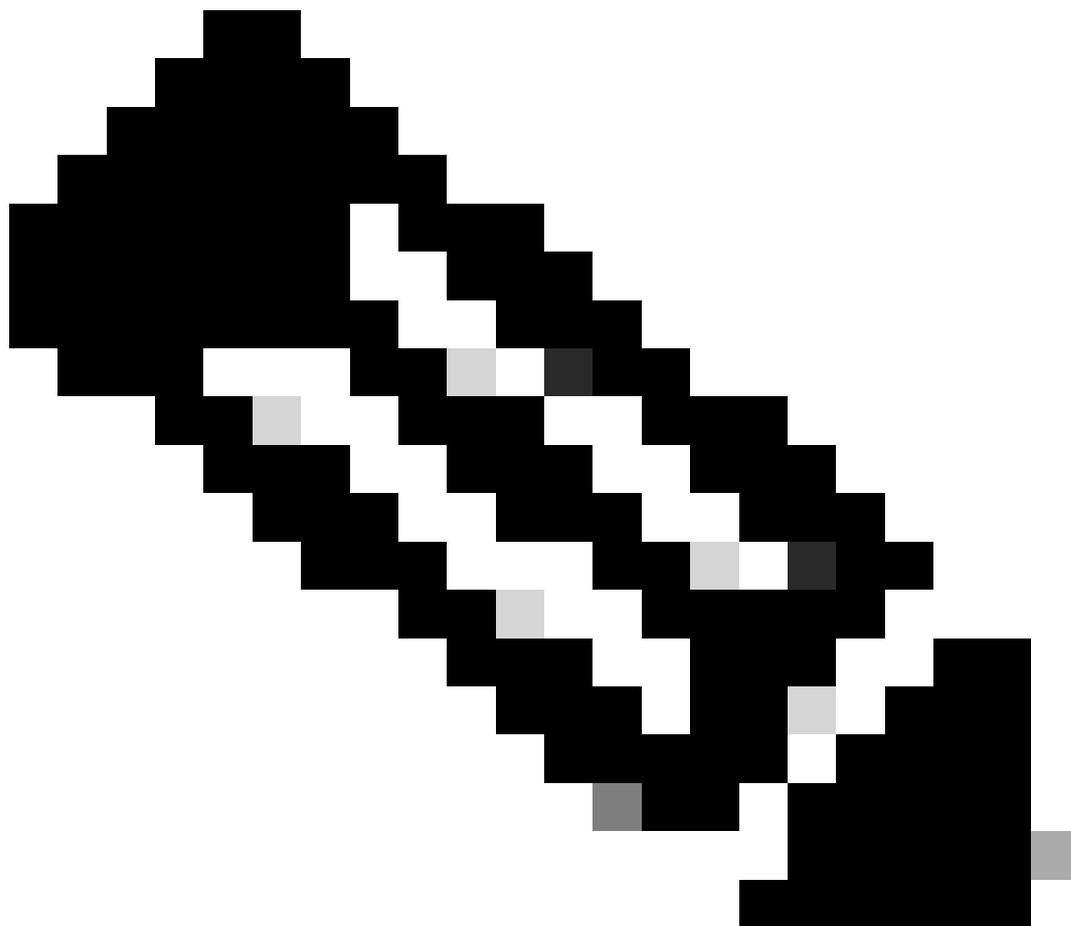
```

> Frame 12568: 188 bytes on wire (1504 bits), 188 bytes captured (1504 bits)
> Ethernet II, Src: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff), Dst: VMware_8d:f3:64 (00:50:56:8d:f3:64)
> Internet Protocol Version 4, Src: 10.61.70.23, Dst: 10.48.48.185
> Transmission Control Protocol, Src Port: 65238, Dst Port: 3128, Seq: 1, Ack: 1, Len: 122
< Hypertext Transfer Protocol
  < GET http://example.com/ HTTP/1.1\r\n
    > [Expert Info (Chat/Sequence): GET http://example.com/ HTTP/1.1\r\n
      Request Method: GET
      Request URI: http://example.com/
      Request Version: HTTP/1.1
      Host: example.com\r\n
      User-Agent: curl/8.4.0\r\n
      Accept: */*\r\n
      Proxy-Connection: Keep-Alive\r\n
      \r\n
      [Full request URI: http://example.com/]
      [HTTP request 1/1]
      [Response in frame: 12852]
  
```

Image - Client an SWA HTTP GET - explicit

Dieser Wert stellt den gesamten Datenverkehrsfluss vom Client zum SWA, dann zum Webserver und schließlich zurück zum Client dar.

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
12544	2024-01-25 09:35:25.989719	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	78	2	65238 → 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=1762371849 TSecr=0 SACK_PERM
12545	2024-01-25 09:35:25.989748	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	74	2	3128 → 65238 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1360 WS=64 SACK_PERM TSval=322700883 TSecr=1762371849
12567	2024-01-25 09:35:26.046546	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=1 Ack=1 Win=132288 Len=0 TSval=1762371848 TSecr=322700883
12568	2024-01-25 09:35:26.046877	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	HTTP	188	2	GET http://example.com/ HTTP/1.1
12569	2024-01-25 09:35:26.046945	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [ACK] Seq=1 Ack=123 Win=65408 Len=0 TSval=3227008847 TSecr=1762371849
12570	2024-01-25 09:35:26.053195	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	74	3	23146 → 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1360 WS=64 SACK_PERM TSval=3190021713 TSecr=0
12778	2024-01-25 09:35:26.168035	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	74	3	80 → 23146 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1360 SACK_PERM TSval=2163592063 TSecr=172700883
12779	2024-01-25 09:35:26.168077	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=1 Ack=1 Win=13568 Len=0 TSval=3190021832 TSecr=2163592063
12780	2024-01-25 09:35:26.168172	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	HTTP	242	3	GET / HTTP/1.1
12833	2024-01-25 09:35:26.280446	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	3	80 → 23146 [ACK] Seq=1 Ack=177 Win=67072 Len=0 TSval=2163592176 TSecr=3190021832
12834	2024-01-25 09:35:26.281757	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	1414	3	80 → 23146 [ACK] Seq=1 Ack=177 Win=67072 Len=1348 TSval=2163592177 TSecr=3190021832 [TCP seq
12835	2024-01-25 09:35:26.281789	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=177 Ack=1349 Win=12224 Len=0 TSval=3190021942 TSecr=2163592177
12836	2024-01-25 09:35:26.281793	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	HTTP	325	3	HTTP/1.1 200 OK (text/html)
12837	2024-01-25 09:35:26.281801	10.48.48.185	VWware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	23146 → 80 [ACK] Seq=177 Ack=1608 Win=11968 Len=0 TSval=3190021942 TSecr=2163592177
12851	2024-01-25 09:35:26.286288	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	1254	2	3128 → 65238 [ACK] Seq=1 Ack=123 Win=65408 Len=1188 TSval=3227001086 TSecr=1762371849 [TCP s
12852	2024-01-25 09:35:26.286297	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	HTTP	599	2	HTTP/1.1 200 OK (text/html)
12992	2024-01-25 09:35:26.347113	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=123 Ack=189 Win=131072 Len=0 TSval=1762372145 TSecr=3227001086
12993	2024-01-25 09:35:26.347815	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=123 Ack=1722 Win=130560 Len=0 TSval=1762372145 TSecr=3227001086
12994	2024-01-25 09:35:26.353174	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	2	65238 → 3128 [FIN, ACK] Seq=123 Ack=1722 Win=131072 Len=0 TSval=1762372150 TSecr=3227001086
12995	2024-01-25 09:35:26.353217	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [ACK] Seq=1722 Ack=124 Win=65408 Len=0 TSval=3227001147 TSecr=1762372150
12996	2024-01-25 09:35:26.353397	10.48.48.185	VWware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 65238 [FIN, ACK] Seq=1722 Ack=124 Win=65408 Len=0 TSval=3227001147 TSecr=1762372150
12997	2024-01-25 09:35:26.412438	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	VWware_8d:f3:64	TCP	66	2	65238 → 3128 [ACK] Seq=124 Ack=1723 Win=131072 Len=0 TSval=1762372212 TSecr=3227001147



Hinweis: Jeder Datenverkehrsstrom zeichnet sich durch eine andere Farbe aus. Der Datenverkehrsfluss vom Client zum SWA ist eine Farbe, der Datenverkehrsfluss vom SWA zum Webserver eine andere.

Time	10.61.70.23	10.48.48.185	93.184.216.34	Comment
2024-01-25 09:35:25.989719	65238 → 3128 [SYN] Seq=0 Win=65535 Len=0	3128		TCP: 65238 → 3128 [SYN] Seq=0 Win=65535 ...
2024-01-25 09:35:25.989748	65238 → 3128 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0	3128		TCP: 3128 → 65238 [SYN, ACK] Seq=0 Ack=1 ...
2024-01-25 09:35:26.046546	65238 → 3128 [ACK] Seq=1 Ack=1 Win=13228 Len=0	3128		TCP: 65238 → 3128 [ACK] Seq=1 Ack=1 Win=1...
2024-01-25 09:35:26.046877	65238 → 3128 GET http://example.com/ HTTP/1.1	3128		HTTP: GET http://example.com/ HTTP/1.1
2024-01-25 09:35:26.046945	65238 → 3128 [ACK] Seq=1 Ack=123 Win=654 Len=0	3128		TCP: 3128 → 65238 [ACK] Seq=1 Ack=123 Win...
2024-01-25 09:35:26.053195	23146 → 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1360 WS=64 SACK_PERM TSval=3191296932 TSecr=0	80		TCP: 23146 → 80 [SYN] Seq=0 Win=12288 Le...
2024-01-25 09:35:26.168035	80 → 23146 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=811197678 TSecr=0	23146		TCP: 80 → 23146 [SYN, ACK] Seq=0 Ack=1 Wi...
2024-01-25 09:35:26.168077	23146 → 80 [ACK] Seq=1 Ack=1 Win=13568 Len=0	80		TCP: 23146 → 80 [ACK] Seq=1 Ack=1 Win=135...
2024-01-25 09:35:26.168172	23146 → 80 GET / HTTP/1.1	80		HTTP: GET / HTTP/1.1
2024-01-25 09:35:26.280446	80 → 23146 [ACK] Seq=1 Ack=177 Win=65535 Len=0	23146		TCP: 80 → 23146 [ACK] Seq=1 Ack=177 Win=6...
2024-01-25 09:35:26.281757	80 → 23146 [ACK] Seq=1 Ack=177 Win=65535 Len=0	23146		TCP: 80 → 23146 [ACK] Seq=1 Ack=177 Win=6...
2024-01-25 09:35:26.281789	23146 → 80 [ACK] Seq=177 Ack=1349 Win=12 Len=0	80		TCP: 23146 → 80 [ACK] Seq=177 Ack=1349 Wl...
2024-01-25 09:35:26.281793	23146 → 80 HTTP/1.1 200 OK (text/html)	80		HTTP: HTTP/1.1 200 OK (text/html)
2024-01-25 09:35:26.281801	23146 → 80 [ACK] Seq=177 Ack=1608 Win=11 Len=0	80		TCP: 23146 → 80 [ACK] Seq=177 Ack=1608 Wl...
2024-01-25 09:35:26.286288	65238 → 3128 [ACK] Seq=1 Ack=123 Win=654 Len=0	3128		TCP: 3128 → 65238 [ACK] Seq=1 Ack=123 Win...
2024-01-25 09:35:26.286297	65238 → 3128 HTTP/1.1 200 OK (text/html)	3128		HTTP: HTTP/1.1 200 OK (text/html)
2024-01-25 09:35:26.347713	65238 → 3128 [ACK] Seq=123 Ack=1189 Win=0 Len=0	3128		TCP: 65238 → 3128 [ACK] Seq=123 Ack=1189 ...
2024-01-25 09:35:26.347815	65238 → 3128 [ACK] Seq=123 Ack=1722 Win=0 Len=0	3128		TCP: 65238 → 3128 [ACK] Seq=123 Ack=1722 ...
2024-01-25 09:35:26.353174	65238 → 3128 [FIN, ACK] Seq=123 Ack=1722 Win=0 Len=0	3128		TCP: 65238 → 3128 [FIN, ACK] Seq=123 Ack=1...
2024-01-25 09:35:26.353217	65238 → 3128 [ACK] Seq=1722 Ack=124 Win=0 Len=0	3128		TCP: 3128 → 65238 [ACK] Seq=1722 Ack=124 ...
2024-01-25 09:35:26.353397	65238 → 3128 [FIN, ACK] Seq=1722 Ack=124 Win=0 Len=0	3128		TCP: 3128 → 65238 [FIN, ACK] Seq=1722 Ack...
2024-01-25 09:35:26.412438	65238 → 3128 [ACK] Seq=124 Ack=1723 Win=0 Len=0	3128		TCP: 65238 → 3128 [ACK] Seq=124 Ack=1723 ...

Image - Datenverkehrsfluss HTTP Explicit - kein Cache

Beispiel für AccessLogs:

1706172876.686 224 10.61.70.23 TCP_MISS/200 1721 GET http://www.example.com/ - DIRECT/www.example.com t

Datenverkehr mit zwischengespeicherten Daten

Dieser Wert stellt den gesamten Datenverkehrsfluss vom Client zum SWA dar, wenn sich die Daten im SWA-Cache befinden.

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
1920	2024-01-25 09:56:41.209030	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	78	2	55709 → 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=3417110271 TSecr=0 SACK_PERM
1921	2024-01-25 09:56:41.209111	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	74	2	3128 → 55709 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1360 WS=64 SACK_PERM TSval=3687923930 TSecr=0
1922	2024-01-25 09:56:41.265937	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	2	55709 → 3128 [ACK] Seq=1 Ack=1 Win=13228 Len=0 TSval=3417110333 TSecr=3687923930
1923	2024-01-25 09:56:41.266065	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	HTTP	188	2	GET http://example.com/ HTTP/1.1
1924	2024-01-25 09:56:41.266114	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 55709 [ACK] Seq=1 Ack=123 Win=65856 Len=0 TSval=3687923930 TSecr=3417110333
1925	2024-01-25 09:56:41.269061	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	74	3	16088 → 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1360 WS=64 SACK_PERM TSval=3191296932 TSecr=0
1943	2024-01-25 09:56:41.385086	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	74	3	80 → 16088 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=811197678 TSecr=0
1944	2024-01-25 09:56:41.385174	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	16088 → 80 [ACK] Seq=1 Ack=1 Win=13568 Len=0 TSval=3191297843 TSecr=811197678
1945	2024-01-25 09:56:41.385270	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	HTTP	292	3	GET / HTTP/1.1
1946	2024-01-25 09:56:41.509528	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	3	80 → 16088 [ACK] Seq=1 Ack=227 Win=67072 Len=0 TSval=811197793 TSecr=3191297843
1947	2024-01-25 09:56:41.518195	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	HTTP	365	3	HTTP/1.1 304 Not Modified
1948	2024-01-25 09:56:41.518259	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	16088 → 80 [ACK] Seq=227 Ack=300 Win=13248 Len=0 TSval=3191297172 TSecr=811197793
1949	2024-01-25 09:56:41.518429	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	16088 → 80 [FIN, ACK] Seq=227 Ack=300 Win=13568 Len=0 TSval=3191297172 TSecr=811197793
1972	2024-01-25 09:56:41.513099	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	1254	2	3128 → 55709 [ACK] Seq=1 Ack=123 Win=65856 Len=1188 TSval=3687924179 TSecr=3417110333 [TCP ...
1973	2024-01-25 09:56:41.513111	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	HTTP	599	2	HTTP/1.1 200 OK (text/html)
1974	2024-01-25 09:56:41.585807	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	2	55709 → 3128 [ACK] Seq=123 Ack=1189 Win=131072 Len=0 TSval=3417110640 TSecr=3687924179
1975	2024-01-25 09:56:41.600259	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	2	55709 → 3128 [ACK] Seq=123 Ack=1722 Win=130560 Len=0 TSval=3417110649 TSecr=3687924179
1976	2024-01-25 09:56:41.604113	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	2	55709 → 3128 [FIN, ACK] Seq=123 Ack=1722 Win=131072 Len=0 TSval=3417110652 TSecr=3687924179
1977	2024-01-25 09:56:41.604191	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 55709 [ACK] Seq=1722 Ack=124 Win=65856 Len=0 TSval=3687924269 TSecr=3417110652
1978	2024-01-25 09:56:41.604293	10.48.48.185	Vmware_8d:f3:64	10.61.70.23	Cisco_9d:b9:ff	TCP	66	2	3128 → 55709 [FIN, ACK] Seq=1722 Ack=124 Win=65856 Len=0 TSval=3687924269 TSecr=3417110652
1979	2024-01-25 09:56:41.630731	93.184.216.34	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	3	80 → 16088 [FIN, ACK] Seq=228 Ack=228 Win=67072 Len=0 TSval=811197917 TSecr=3191297172
1980	2024-01-25 09:56:41.636832	10.48.48.185	Vmware_8d:f3:64	93.184.216.34	Cisco_9d:b9:ff	TCP	66	3	16088 → 80 [ACK] Seq=228 Ack=301 Win=13568 Len=0 TSval=3191297302 TSecr=811197917
1981	2024-01-25 09:56:41.662464	10.61.70.23	Cisco_9d:b9:ff	10.48.48.185	Vmware_8d:f3:64	TCP	66	2	55709 → 3128 [ACK] Seq=124 Ack=1723 Win=131072 Len=0 TSval=3417110729 TSecr=3687924269

Bild - Explizit zwischengespeicherte HTTP-Daten



Hinweis: Wie Sie sehen, gibt der Webserver die HTTP-Antwort 304: Cache not Modified (Cache nicht geändert) zurück. (in diesem Beispiel Paketnummer 1947)

Time	10.61.70.23	10.48.48.185	93.184.216.34	Comment
2024-01-25 09:56:41.209030	55709	55709 → 3128 [SYN] Seq=0 Win=65535 Len=...	3128	TCP: 55709 → 3128 [SYN] Seq=0 Win=65535 ...
2024-01-25 09:56:41.209111	55709	3128 → 55709 [SYN, ACK] Seq=0 Ack=1 Win=6...	3128	TCP: 3128 → 55709 [SYN, ACK] Seq=0 Ack=1 ...
2024-01-25 09:56:41.265937	55709	55709 → 3128 [ACK] Seq=1 Ack=1 Win=13228...	3128	TCP: 55709 → 3128 [ACK] Seq=1 Ack=1 Win=1...
2024-01-25 09:56:41.266065	55709	GET http://example.com/ HTTP/1.1	3128	HTTP: GET http://example.com/ HTTP/1.1
2024-01-25 09:56:41.266114	55709	3128 → 55709 [ACK] Seq=1 Ack=123 Win=658...	3128	TCP: 3128 → 55709 [ACK] Seq=1 Ack=123 Win...
2024-01-25 09:56:41.269061		16088 → 80 [SYN] Seq=0 Win=12288 Len=0 M...	80	TCP: 16088 → 80 [SYN] Seq=0 Win=12288 Le...
2024-01-25 09:56:41.385086		80 → 16088 [SYN, ACK] Seq=0 Ack=1 Win=65...	80	TCP: 80 → 16088 [SYN, ACK] Seq=0 Ack=1 Wi...
2024-01-25 09:56:41.385174		16088 → 80 [ACK] Seq=1 Ack=1 Win=13568 L...	80	TCP: 16088 → 80 [ACK] Seq=1 Ack=1 Win=135...
2024-01-25 09:56:41.385270		GET / HTTP/1.1	80	HTTP: GET / HTTP/1.1
2024-01-25 09:56:41.509528		80 → 16088 [ACK] Seq=1 Ack=227 Win=67072...	80	TCP: 80 → 16088 [ACK] Seq=1 Ack=227 Win...
2024-01-25 09:56:41.510195		HTTP/1.1 304 Not Modified	80	HTTP: HTTP/1.1 304 Not Modified
2024-01-25 09:56:41.510259		16088 → 80 [ACK] Seq=227 Ack=300 Win=132...	80	TCP: 16088 → 80 [ACK] Seq=227 Ack=300 Wi...
2024-01-25 09:56:41.510429		16088 → 80 [FIN, ACK] Seq=227 Ack=300 Win...	80	TCP: 16088 → 80 [FIN, ACK] Seq=227 Ack=30...
2024-01-25 09:56:41.513099	55709	3128 → 55709 [ACK] Seq=1 Ack=123 Win=658...	3128	TCP: 3128 → 55709 [ACK] Seq=1 Ack=123 Win...
2024-01-25 09:56:41.513111	55709	HTTP/1.1 200 OK (text/html)	3128	HTTP: HTTP/1.1 200 OK (text/html)
2024-01-25 09:56:41.585507	55709	55709 → 3128 [ACK] Seq=123 Ack=1189 Win...	3128	TCP: 55709 → 3128 [ACK] Seq=123 Ack=1189 ...
2024-01-25 09:56:41.600269	55709	55709 → 3128 [ACK] Seq=123 Ack=1722 Win...	3128	TCP: 55709 → 3128 [ACK] Seq=123 Ack=1722 ...
2024-01-25 09:56:41.604113	55709	55709 → 3128 [FIN, ACK] Seq=123 Ack=1722...	3128	TCP: 55709 → 3128 [FIN, ACK] Seq=123 Ack=1...
2024-01-25 09:56:41.604191	55709	3128 → 55709 [ACK] Seq=1722 Ack=124 Win...	3128	TCP: 3128 → 55709 [ACK] Seq=1722 Ack=124 ...
2024-01-25 09:56:41.604293	55709	3128 → 55709 [FIN, ACK] Seq=1722 Ack=124...	3128	TCP: 3128 → 55709 [FIN, ACK] Seq=1722 Ack=...
2024-01-25 09:56:41.636731		16088 → 80 [FIN, ACK] Seq=300 Ack=228 Win...	80	TCP: 80 → 16088 [FIN, ACK] Seq=300 Ack=22...
2024-01-25 09:56:41.636832		16088 → 80 [ACK] Seq=228 Ack=301 Win=135...	80	TCP: 16088 → 80 [ACK] Seq=228 Ack=301 Wi...
2024-01-25 09:56:41.662464	55709	55709 → 3128 [ACK] Seq=124 Ack=1723 Win...	3128	TCP: 55709 → 3128 [ACK] Seq=124 Ack=1723 ...

Image - Flow HTTP explizit mit Cache

Nachfolgend finden Sie ein Beispiel für die HTTP-Antwort 304.

```
> Frame 1947: 365 bytes on wire (2920 bits), 365 bytes captured (2920 bits)
> Ethernet II, Src: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff), Dst: VMware_8d:f3:64 (00:50:56:8d:f3:64)
> Internet Protocol Version 4, Src: 93.184.216.34, Dst: 10.48.48.185
> Transmission Control Protocol, Src Port: 80, Dst Port: 16088, Seq: 1, Ack: 227, Len: 299
< Hypertext Transfer Protocol
  < HTTP/1.1 304 Not Modified\r\n
    < [Expert Info (Chat/Sequence): HTTP/1.1 304 Not Modified\r\n]
      [HTTP/1.1 304 Not Modified\r\n]
      [Severity level: Chat]
      [Group: Sequence]
      Response Version: HTTP/1.1
      Status Code: 304
      [Status Code Description: Not Modified]
      Response Phrase: Not Modified
      Accept-Ranges: bytes\r\n
      Age: 519756\r\n
      Cache-Control: max-age=604800\r\n
      Date: Thu, 25 Jan 2024 08:57:08 GMT\r\n
      Etag: "3147526947"\r\n
      Expires: Thu, 01 Feb 2024 08:57:08 GMT\r\n
      Last-Modified: Thu, 17 Oct 2019 07:18:26 GMT\r\n
      Server: ECS (dce/2694)\r\n
      Vary: Accept-Encoding\r\n
      X-Cache: HIT\r\n
      \r\n
      [HTTP response 1/1]
      [Time since request: 0.124925000 seconds]
      [Request in frame: 1945]
      [Request URI: http://example.com/]
```

Bild - explizite HTTP-304-Antwort

Beispiel für AccessLogs:

```
1706173001.489 235 10.61.70.23 TCP_REFRESH_HIT/200 1721 GET http://www.example.com/ - DIRECT/www.examp1
```

HTTP-Datenverkehr bei expliziter Bereitstellung ohne Authentifizierung

Client und SWA

Der Netzwerkverkehr verläuft zwischen der IP-Adresse des Clients und der IP-Adresse der SWA-Proxyschnittstelle (in der Regel handelt es sich um die P1-Schnittstelle, kann aber auch die P2- oder die Management-Schnittstelle sein, je nach Proxy-Konfiguration).

Der Datenverkehr vom Client ist für TCP-Port 80 oder 3128 zum SWA bestimmt (Standard-SWA-Proxyports sind TCP 80 und 3128, in diesem Beispiel verwenden wir Port 3128).

- TCP-Handshake

- HTTP CONNECT vom Client (Ziel-IP = SWA, Ziel-Port = 3128)
- HTTP-Antwort vom Proxy (Quell-IP = SWA)
- Client Hello mit SNI der URL (Quell-IP = Client)
- Server Hello (Quell-IP = SWA)
- Server Key Exchange (Quell-IP = SWA)
- Client Key Exchange (Quell-IP = Client)
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
18	2024-01-25 12:31:37.318168644	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	78	12	61484 -> 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=1676451324 TSecr=0 SACK_PERM
19	2024-01-25 12:31:37.338015315	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	74	12	3128 -> 61484 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM TSval=441495437
20	2024-01-25 12:31:37.370297760	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1 Ack=1 Win=132288 Len=0 TSval=1676451392 TSecr=441495437
21	2024-01-25 12:31:37.383167	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	HTTP	277	12	CONNECT example.com:443 HTTP/1.1
22	2024-01-25 12:31:37.324946619	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=1 Ack=212 Win=65344 Len=0 TSval=441495507 TSecr=1676451392
26	2024-01-25 12:31:38.731815	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	HTTP	185	12	HTTP/1.1 200 Connection established
27	2024-01-25 12:31:38.388877561	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=212 Ack=40 Win=132224 Len=0 TSval=1676451630 TSecr=441495677
28	2024-01-25 12:31:38.322347166	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TLSv1.2	715	12	Client Hello (SNI=example.com)
29	2024-01-25 12:31:38.182072475	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=40 Ack=861 Win=64784 Len=0 TSval=441495747 TSecr=1676451630
49	2024-01-25 12:31:38.282097668	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	1254	12	Server Hello
50	2024-01-25 12:31:38.153429867	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	1254	12	Certificate
51	2024-01-25 12:31:38.965425	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	190	12	Server Key Exchange, Server Hello Done
54	2024-01-25 12:31:38.824826	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=861 Ack=1228 Win=131008 Len=0 TSval=1676452189 TSecr=441496237
55	2024-01-25 12:31:38.344661913	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=861 Ack=2540 Win=129728 Len=0 TSval=1676452189 TSecr=441496237
56	2024-01-25 12:31:38.173832950	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TLSv1.2	159	12	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
57	2024-01-25 12:31:38.422856787	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=2540 Ack=954 Win=64640 Len=0 TSval=441496317 TSecr=1676452193
58	2024-01-25 12:31:38.244514147	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	117	12	Change Cipher Spec, Encrypted Handshake Message
59	2024-01-25 12:31:38.328702336	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=954 Ack=2591 Win=131008 Len=0 TSval=1676452265 TSecr=441496317
60	2024-01-25 12:31:38.151248214	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TLSv1.2	562	12	Application Data
61	2024-01-25 12:31:38.257435452	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=2591 Ack=1450 Win=64192 Len=0 TSval=441496387 TSecr=1676452265
82	2024-01-25 12:31:39.165886323	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	112	12	Application Data
83	2024-01-25 12:31:39.342008	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1450 Ack=2637 Win=131008 Len=0 TSval=1676452764 TSecr=441496807
84	2024-01-25 12:31:39.280484740	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	1209	12	Application Data, Application Data
85	2024-01-25 12:31:39.128618294	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1450 Ack=3780 Win=129920 Len=0 TSval=1676452838 TSecr=441496887
86	2024-01-25 12:31:39.092047	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TLSv1.2	497	12	Application Data
87	2024-01-25 12:31:39.277889790	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=3780 Ack=1881 Win=63008 Len=0 TSval=441496997 TSecr=1676452884
94	2024-01-25 12:31:39.126123713	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	119	12	Application Data
95	2024-01-25 12:31:39.688580	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1881 Ack=3833 Win=131008 Len=0 TSval=1676453324 TSecr=441497377
96	2024-01-25 12:31:39.288575172	10.48.48.165	VWware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLSv1.2	1192	12	Application Data, Application Data
97	2024-01-25 12:31:39.295531248	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1881 Ack=4959 Win=129920 Len=0 TSval=1676453397 TSecr=441497447
150	2024-01-25 12:31:49.143134836	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VWware_8d:9a:f4	TCP	60	12	[TCP Keep-Alive] 61484 -> 3128 [ACK] Seq=1880 Ack=4959 Win=131072 Len=0

Image - HTTPS-Client an SWA-Explicit - kein Cache

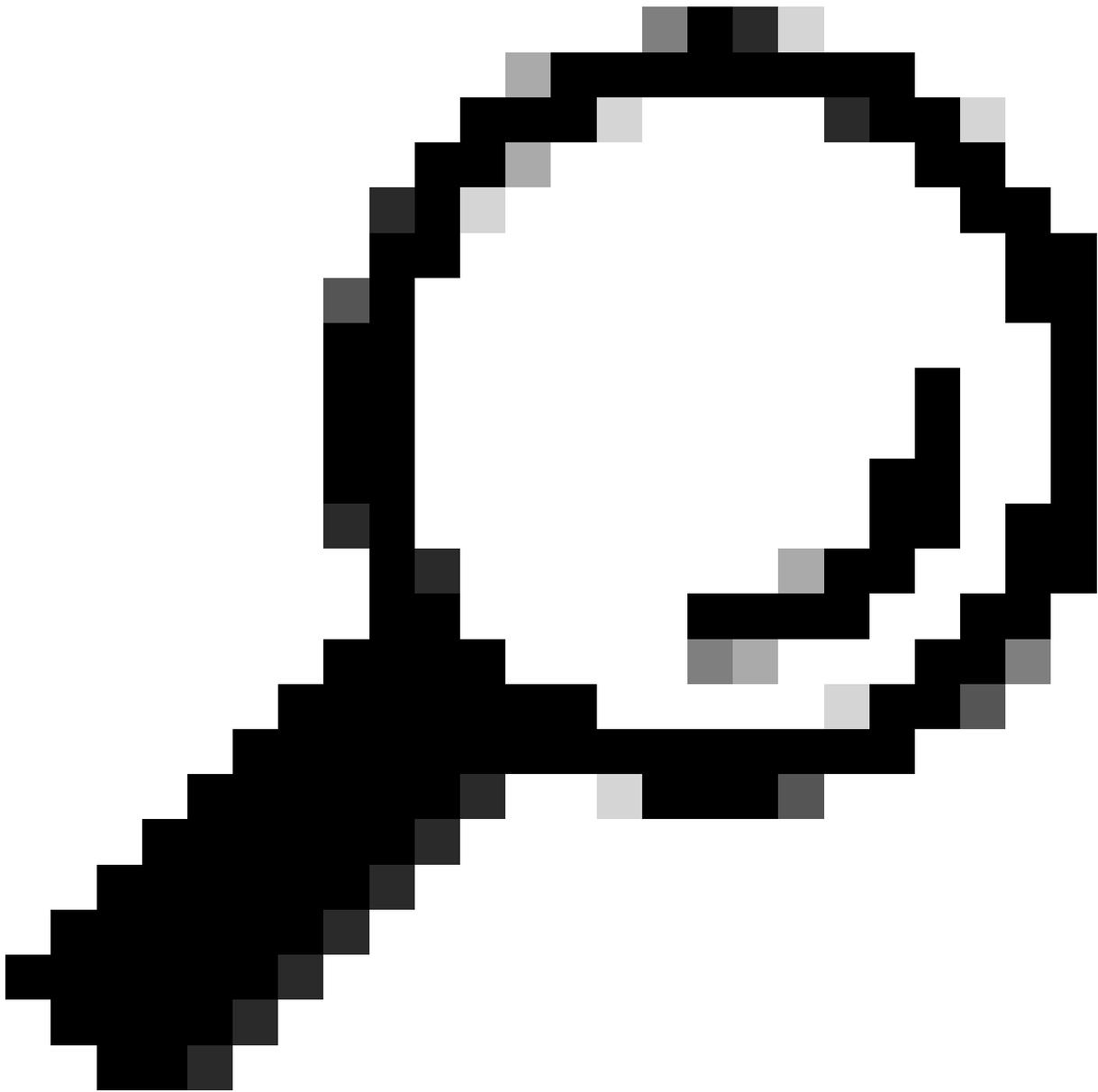
Hier sind Details von Client Hello von Client an SWA, wie Sie in der Server Name Indication (SNI) die URL des Webservers sehen können, die in diesem Beispiel www.example.com ist und von dem Client 17 Cipher Suites angekündigt wurde:

```

> Frame 28: 715 bytes on wire (5720 bits), 715 bytes captured (5720 bits)
> Ethernet II, Src: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff), Dst: VMware_8d:9a:f4 (00:50:56:8d:9a:f4)
> Internet Protocol Version 4, Src: 10.61.70.23, Dst: 10.48.48.165
> Transmission Control Protocol, Src Port: 61484, Dst Port: 3128, Seq: 212, Ack: 40, Len: 649
< Hypertext Transfer Protocol
  [Proxy-Connect-Hostname: example.com]
  [Proxy-Connect-Port: 443]
< Transport Layer Security
  < TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 644
  < Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 640
    Version: TLS 1.2 (0x0303)
  > Random: 8f2d33b577f5cd05ab284c0a64a929e5dd29c940aa73ccc3f4bcfaf8509078d
    Session ID Length: 32
    Session ID: e91649fe756a373ce70f5b65c9729b805d864f8f39ac783b2feb9a49ced7de6b
    Cipher Suites Length: 34
  > Cipher Suites (17 suites) ←
    Compression Methods Length: 1
  > Compression Methods (1 method)
    Extensions Length: 533
  < Extension: server_name (len=16) name=example.com
    Type: server_name (0)
    Length: 16
  < Server Name Indication extension
    Server Name list length: 14
    Server Name Type: host_name (0)
    Server Name length: 11
    Server Name: example.com
  > Extension: extended_master_secret (len=0)
  > Extension: renegotiation_info (len=1)
  > Extension: supported_groups (len=14)
  > Extension: ec_point_formats (len=2)
  > Extension: application_layer_protocol_negotiation (len=14)
  > Extension: status_request (len=5)
  > Extension: delegated_credentials (len=10)
  > Extension: key_share (len=107) x25519, secp256r1
  > Extension: supported_versions (len=5) TLS 1.3, TLS 1.2
  > Extension: signature_algorithms (len=24)
  > Extension: record_size_limit (len=2)
  > Extension: encrypted_client_hello (len=281)
    JA4: t13d1713h2 5h57614c22h0 748f4c70de1c1

```

Bild - HTTPS-Client hello - Explizit - Client an SWA



Tipp: Sie können diesen Filter in Wireshark verwenden, um nach URL/SNI zu suchen:
`tls.handshake.extensions_server_name == "www.example.com"`

Hier ist ein Beispiel für ein Zertifikat, das SWA an den Client gesendet hat:

```

> Frame 50: 1254 bytes on wire (10032 bits), 1254 bytes captured (10032 bits)
> Ethernet II, Src: VMware_Bd:9a:f4 (00:50:56:8d:9a:f4), Dst: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff)
> Internet Protocol Version 4, Src: 10.48.48.165, Dst: 10.61.70.23
> Transmission Control Protocol, Src Port: 3128, Dst Port: 61484, Seq: 1228, Ack: 861, Len: 1188
> [2 Reassembled TCP Segments (2105 bytes): #49(1107), #50(998)]
> Hypertext Transfer Protocol
  [Proxy-Connect-Hostname: example.com]
  [Proxy-Connect-Port: 443]
> Transport Layer Security
  > TLSv1.2 Record Layer: Handshake Protocol: Certificate
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 2100
  > Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 2096
    Certificates Length: 2093
  > Certificates (2093 bytes)
    Certificate Length: 1105
  > Certificate [truncated]: 3082044d30820335a00302010202140279103122f2aad73d32683b716d2a7d4ead7d47300d06092a864886f70d01010b05003047310b300906035504061302555310e300c060355040a1.
  > signedCertificate
    version: v3 (2)
    serialNumber: 0x0279103122f2aad73d32683b716d2a7d4ead7d47
  > signature (sha256WithRSAEncryption)
  > issuer: rdnsSequence (0)
  > rdnsSequence: 4 items (id-at-commonName=CISCO LAB Explicit, id-at-organizationalUnitName=IT, id-at-organizationName=Cisco, id-at-countryName=US)
    > RDNSequence item: 1 item (id-at-countryName=US)
      > RelativeDistinguishedName item (id-at-countryName=US)
        Object Id: 2.5.4.6 (id-at-countryName)
        CountryName: US
    > RDNSequence item: 1 item (id-at-organizationName=Cisco)
      > RelativeDistinguishedName item (id-at-organizationName=Cisco)
        Object Id: 2.5.4.10 (id-at-organizationName)
        > DirectoryString: printableString (1)
          printableString: Cisco
    > RDNSequence item: 1 item (id-at-organizationalUnitName=IT)
      > RelativeDistinguishedName item (id-at-organizationalUnitName=IT)
        Object Id: 2.5.4.11 (id-at-organizationalUnitName)
        > DirectoryString: printableString (1)
          printableString: IT
    > RDNSequence item: 1 item (id-at-commonName=CISCO LAB Explicit)
      > RelativeDistinguishedName item (id-at-commonName=CISCO LAB Explicit)
        Object Id: 2.5.4.3 (id-at-commonName)
        > DirectoryString: printableString (1)
          printableString: CISCO LAB Explicit

```

Image - HTTPS-Zertifikat - Explizit - SWA an Client

SWA und Webserver

Der Netzwerkverkehr erfolgt zwischen der IP-Adresse des Proxys und der IP-Adresse des Webserver.

Der Datenverkehr von SWA ist für TCP-Port 443 bestimmt (nicht für den Proxy-Port)

- TCP-Handshake
- Client Hello (Ziel-IP = Webserver, Ziel-Port = 443)
- Server Hello (Quell-IP = Webserver)
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Length	Stream	Info
23	2024-01-25 12:31:37.383901	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	74	13	24953 → 443 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSval=2549353418 TSecr=0
24	2024-01-25 12:31:38.086918	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	74	13	443 → 24953 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=1727280976 TSecr=0
25	2024-01-25 12:31:38.893381	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=1 Ack=1 Win=12480 Len=0 TSval=2549353558 TSecr=1727280976
30	2024-01-25 12:31:38.358314	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	259	13	Client Hello (SNIM=example.com)
31	2024-01-25 12:31:38.146535406..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	66	13	443 → 24953 [ACK] Seq=1 Ack=194 Win=67072 Len=0 TSval=1727281239 TSecr=2549353688
32	2024-01-25 12:31:38.247031593..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TLSv1.2	1434	13	Server Hello
33	2024-01-25 12:31:38.273349971..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=194 Ack=1369 Win=11136 Len=0 TSval=2549353808 TSecr=1727281240
34	2024-01-25 12:31:38.1341489009..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	1434	13	443 → 24953 [PSH, ACK] Seq=1369 Ack=194 Win=67072 Len=1368 TSval=1727281240 TSecr=254935368
35	2024-01-25 12:31:38.178681044..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=194 Ack=2737 Win=11072 Len=0 TSval=2549353818 TSecr=1727281240
36	2024-01-25 12:31:38.345520	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TLSv1.2	896	13	Certificate, Server Key Exchange, Server Hello Done
37	2024-01-25 12:31:38.161048344..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=194 Ack=3567 Win=10304 Len=0 TSval=2549353818 TSecr=1727281240
38	2024-01-25 12:31:38.062391	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	192	13	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
39	2024-01-25 12:31:38.141402590..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TLSv1.2	117	13	Change Cipher Spec, Encrypted Handshake Message
40	2024-01-25 12:31:38.1809573742..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=320 Ack=3618 Win=12480 Len=0 TSval=2549353988 TSecr=1727281420
64	2024-01-25 12:31:38.1296760748..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	111	13	Application Data
73	2024-01-25 12:31:38.411911657..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	66	13	443 → 24953 [ACK] Seq=3618 Ack=365 Win=67072 Len=0 TSval=1727281896 TSecr=2549354298
74	2024-01-25 12:31:38.3480012513..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	640	13	Application Data, Application Data
78	2024-01-25 12:31:39.283280060..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	66	13	443 → 24953 [ACK] Seq=3618 Ack=939 Win=68096 Len=0 TSval=1727282019 TSecr=2549354468
79	2024-01-25 12:31:39.159843076..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TLSv1.2	1146	13	Application Data, Application Data
80	2024-01-25 12:31:39.1305106563..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=939 Ack=4698 Win=11456 Len=0 TSval=2549354588 TSecr=1727282020
88	2024-01-25 12:31:39.352452851..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	122	13	Application Data
89	2024-01-25 12:31:39.427217571..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	66	13	443 → 24953 [ACK] Seq=4698 Ack=995 Win=68096 Len=0 TSval=1727282552 TSecr=2549354948
90	2024-01-25 12:31:39.1347738670..	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLSv1.2	564	13	Application Data, Application Data
91	2024-01-25 12:31:39.186179736..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TCP	66	13	443 → 24953 [ACK] Seq=4698 Ack=1493 Win=69120 Len=0 TSval=1727282678 TSecr=2549355128
92	2024-01-25 12:31:39.1282826742..	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_Bd:9a:f4	TLSv1.2	1136	13	Application Data, Application Data
93	2024-01-25 12:31:39.048886	10.48.48.165	VMware_Bd:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 → 443 [ACK] Seq=1493 Ack=5768 Win=11264 Len=0 TSval=2549355248 TSecr=1727282680

Bild - HTTPS - explizit - SWA an Webserver

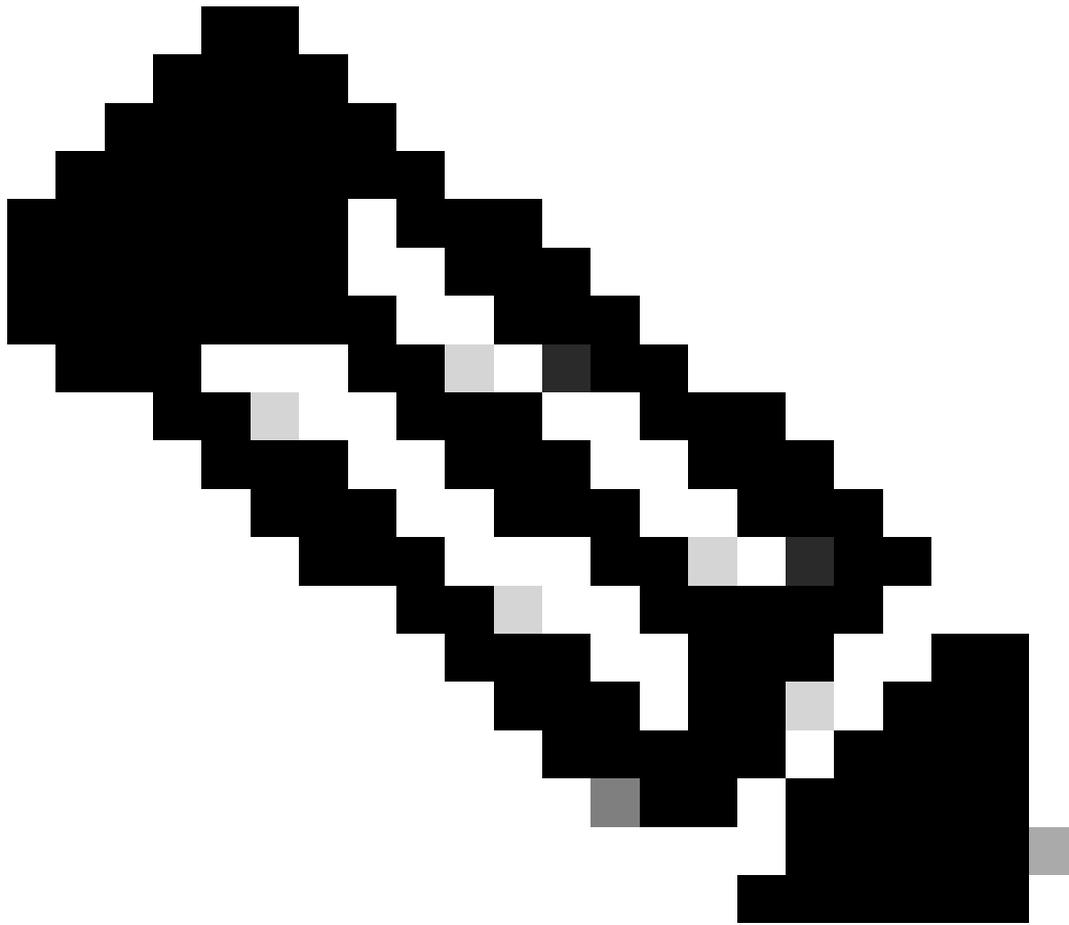
Hier sind die Details von Client Hello von SWA zum Webserver, wie Sie sehen können, dass SWA 12 Cipher Suites angekündigt hat:

```

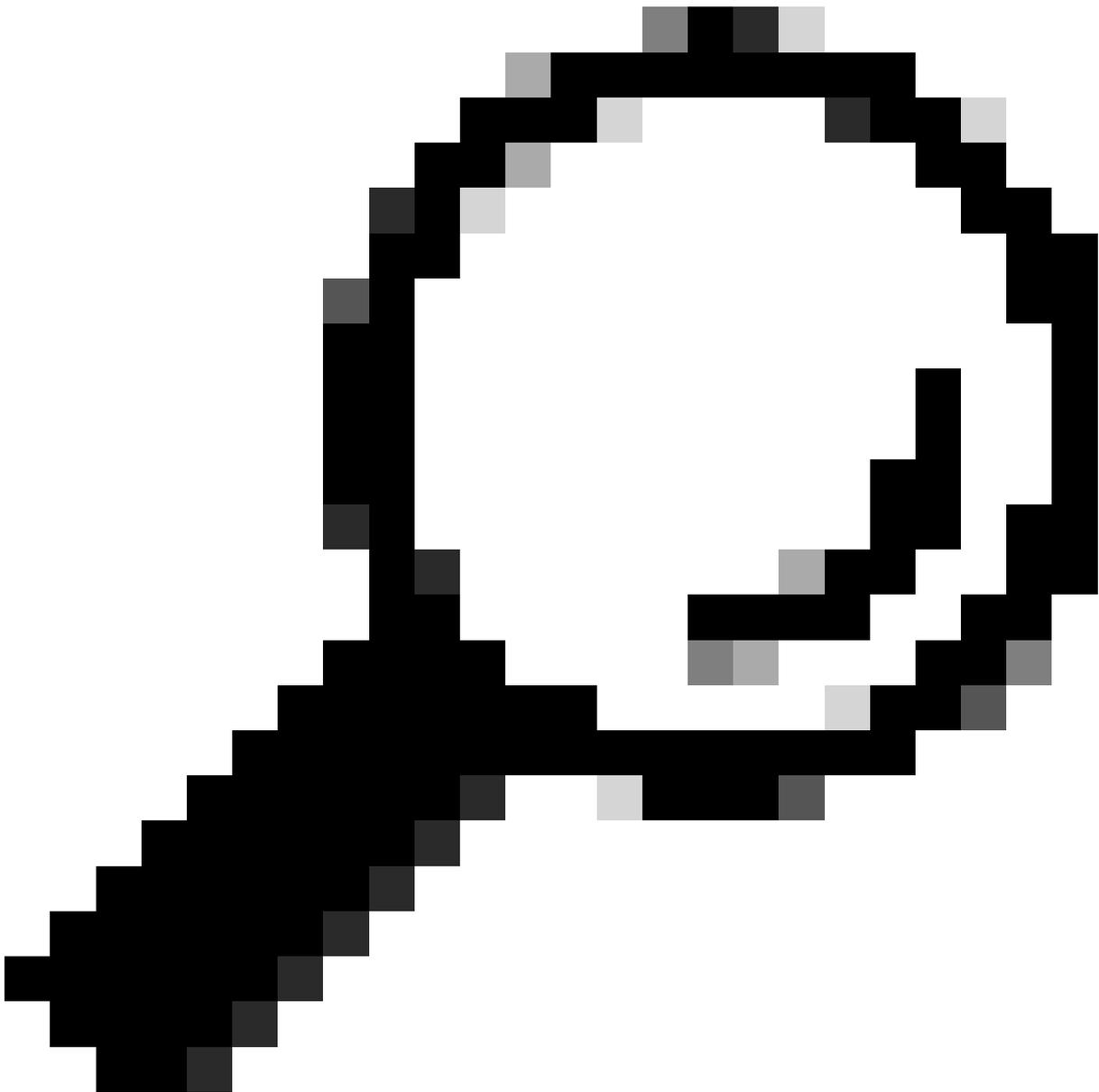
> Frame 30: 259 bytes on wire (2072 bits), 259 bytes captured (2072 bits)
> Ethernet II, Src: VMware_8d:9a:f4 (00:50:56:8d:9a:f4), Dst: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff)
> Internet Protocol Version 4, Src: 10.48.48.165, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 24953, Dst Port: 443, Seq: 1, Ack: 1, Len: 193
▼ Transport Layer Security
  ▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 188
    ▼ Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 184
      Version: TLS 1.2 (0x0303)
      Random: 6601ee708d9db71cf5c7c4584e5facdf08d4de00b208f6d6eb6ade08cc7d3e14
      Session ID Length: 0
      Cipher Suites Length: 24
      Cipher Suites (12 suites) ←
      Compression Methods Length: 1
      Compression Methods (1 method)
      Extensions Length: 119
      ▼ Extension: server_name (len=16) name=example.com
        Type: server_name (0)
        Length: 16
        ▼ Server Name Indication extension
          Server Name list length: 14
          Server Name Type: host_name (0)
          Server Name length: 11
          Server Name: example.com
      > Extension: ec_point_formats (len=4)
      > Extension: supported_groups (len=12)
      > Extension: application_layer_protocol_negotiation (len=11)
      > Extension: encrypt_then_mac (len=0)
      > Extension: extended_master_secret (len=0)
      > Extension: signature_algorithms (len=48)
      [JA4: t12d1207h1_ea129f91df3f_ed727256b201]
      [JA4_r: t12d1207h1_002f,009c,009d,00ff,c009,c013,c02b,c02c,c02f,c030,cca8,cca9_000a,000b,000d,0016,0017_0403,0503,0603,0807,0808,0809,080a,080b,0804,0805,0806,0401,0501,0601,030]
      [JA3 Fullstring: 771,49195-49199-52393-52392-49196-49200-49161-49171-156-157-47-255,0-11-10-16-22-23-13,29-23-30-25-24,0-1-2]
      [JA3: 485a74d85df6d99eb1db31d9c65efe0f]

```

Bild - HTTPS-Client Hello - SWA zum Webserver- Kein Cache



Hinweis: Die hier beobachteten Cipher Suites unterscheiden sich von den Cipher Suites in Client Hello von Client zu SWA, da die SWA, die für die Entschlüsselung dieses Datenverkehrs konfiguriert ist, ihre eigenen Ciphers verwendet.



Tipp: Im Fenster Server Key Exchange (Serverschlüsselaustausch von SWA zu Webserver) wird das Webserverzertifikat angezeigt. Wenn ein Upstreamproxy jedoch eine Konfiguration für Ihren SWA findet, wird sein Zertifikat anstelle des Webserverzertifikats angezeigt.

Hier ist ein Beispiel für HTTP CONNECT vom Client

```

> Frame 21: 277 bytes on wire (2216 bits), 277 bytes captured (2216 bits)
> Ethernet II, Src: Cisco_9d:b9:ff (4c:71:0d:9d:b9:ff), Dst: VMware_8d:9a:f4 (00:50:56:8d:9a:f4)
> Internet Protocol Version 4, Src: 10.61.70.23, Dst: 10.48.48.165
> Transmission Control Protocol, Src Port: 61484, Dst Port: 3128, Seq: 1, Ack: 1, Len: 211
< Hypertext Transfer Protocol
  < CONNECT example.com:443 HTTP/1.1\r\n
    < [Expert Info (Chat/Sequence): CONNECT example.com:443 HTTP/1.1\r\n]
      [CONNECT example.com:443 HTTP/1.1\r\n]
      [Severity level: Chat]
      [Group: Sequence]
      Request Method: CONNECT
      Request URI: example.com:443
      Request Version: HTTP/1.1
      User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:122.0) Gecko/20100101 Firefox/122.0\r\n
      Proxy-Connection: keep-alive\r\n
      Connection: keep-alive\r\n
      Host: example.com:443\r\n
      \r\n
      [Full request URI: example.com:443]
      [HTTP request 1/1]
      [Response in frame: 26]

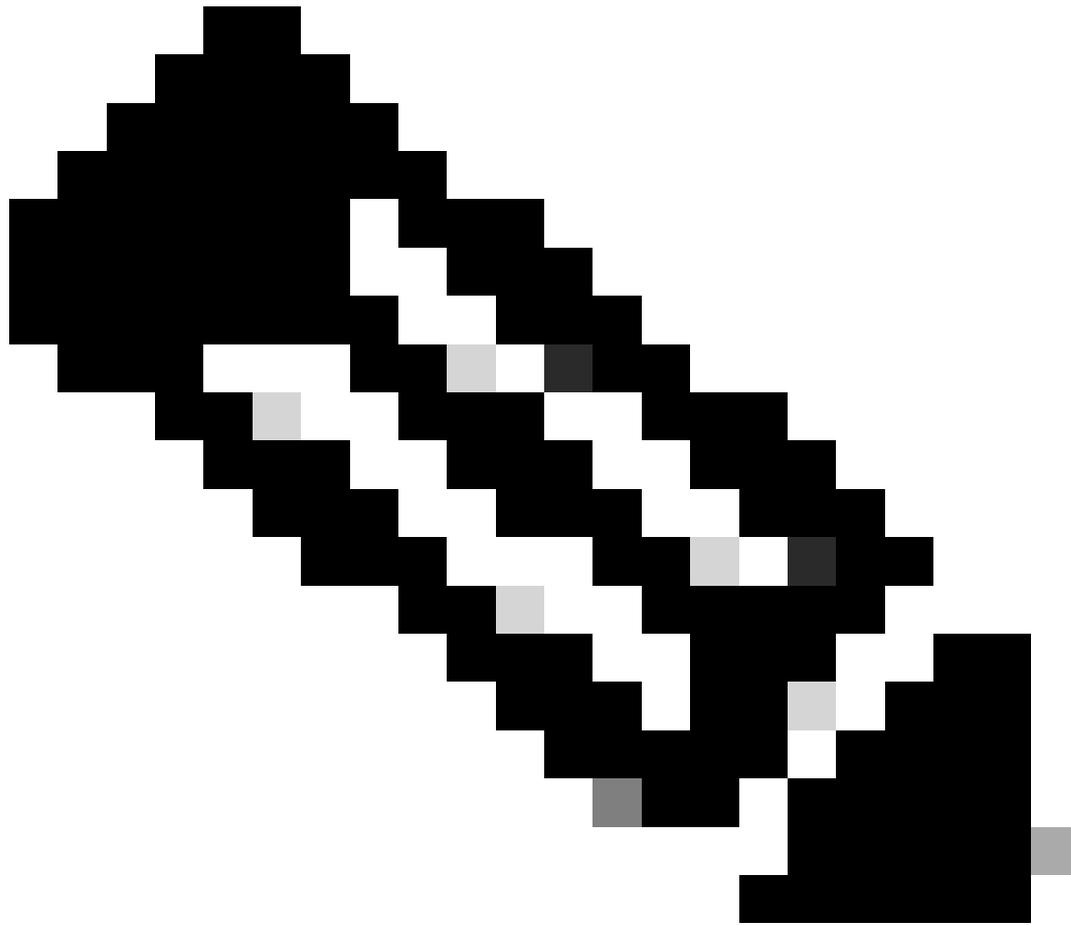
```

Bild - Client HTTP Connect

Dieser Wert stellt den gesamten Datenverkehrsfluss vom Client zum SWA, dann zum Webserver und schließlich zurück zum Client dar.

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Length	Stream	Info
18	2024-01-25 12:31:37.318168644...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	78	12	61484 -> 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=1676451324 TSecr=0 SACK_PERM TSval=441495677 TSecr=0
19	2024-01-25 12:31:37.330015315...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	74	12	3128 -> 61484 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM TSval=441495677 TSecr=0
20	2024-01-25 12:31:37.370297760...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1 Ack=1 Win=132288 Len=0 TSval=1676451392 TSecr=441495437
21	2024-01-25 12:31:37.383167...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	HTTP	277	12	CONNECT example.com:443 HTTP/1.1
22	2024-01-25 12:31:37.324946619...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=1 Ack=212 Win=65344 Len=0 TSval=441495507 TSecr=1676451392
23	2024-01-25 12:31:37.383901...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	74	13	24953 -> 443 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSval=2549353418 TSecr=0
24	2024-01-25 12:31:38.006918...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	74	13	443 -> 24953 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=1727280971 TSecr=0
25	2024-01-25 12:31:38.009381...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=1 Ack=1 Win=12480 Len=0 TSval=2549353558 TSecr=1727280976
26	2024-01-25 12:31:38.731815...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	HTTP	185	12	HTTP/1.1 200 Connection established
27	2024-01-25 12:31:38.380897561...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=212 Ack=40 Win=132224 Len=0 TSval=1676451630 TSecr=441495677
28	2024-01-25 12:31:38.322347166...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	715	12	Client Hello [SNI=example.com]
29	2024-01-25 12:31:38.182072475...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=40 Ack=861 Win=64704 Len=0 TSval=441495747 TSecr=1676451630
30	2024-01-25 12:31:38.350314...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLsv1.2	259	13	Client Hello [SNI=example.com]
31	2024-01-25 12:31:38.146535406...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	13	443 -> 24953 [ACK] Seq=1 Ack=194 Win=67072 Len=0 TSval=1727281239 TSecr=2549353688
32	2024-01-25 12:31:38.247031593...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	1434	13	Server Hello
33	2024-01-25 12:31:38.273349971...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=194 Ack=1369 Win=11136 Len=0 TSval=2549353808 TSecr=1727281240
34	2024-01-25 12:31:38.141480900...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	1434	13	443 -> 24953 [PSH, ACK] Seq=1369 Ack=194 Win=67072 Len=1368 TSval=1727281240 TSecr=2549353808
35	2024-01-25 12:31:38.178681044...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=194 Ack=2737 Win=11072 Len=0 TSval=2549353818 TSecr=1727281240
36	2024-01-25 12:31:38.345520...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	896	13	Certificate, Server Key Exchange, Server Hello Done
37	2024-01-25 12:31:38.161040344...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=194 Ack=3567 Win=10304 Len=0 TSval=2549353818 TSecr=1727281240
38	2024-01-25 12:31:38.062391...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLsv1.2	192	13	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
39	2024-01-25 12:31:38.414028500...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	117	13	Change Cipher Spec, Encrypted Handshake Message
40	2024-01-25 12:31:38.109573742...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=320 Ack=3618 Win=12480 Len=0 TSval=2549353988 TSecr=1727281420
49	2024-01-25 12:31:38.282097660...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	1254	12	Server Hello
50	2024-01-25 12:31:38.115429067...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	1254	12	Certificate
51	2024-01-25 12:31:38.965425...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	190	12	Server Key Exchange, Server Hello Done
54	2024-01-25 12:31:38.824826...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=861 Ack=1228 Win=131008 Len=0 TSval=1676452189 TSecr=441496237
55	2024-01-25 12:31:38.344661913...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=861 Ack=2540 Win=129728 Len=0 TSval=1676452189 TSecr=441496237
56	2024-01-25 12:31:38.173832950...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	159	12	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
57	2024-01-25 12:31:38.422856787...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=2540 Ack=954 Win=64640 Len=0 TSval=441496317 TSecr=1676452193
58	2024-01-25 12:31:38.244514147...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	117	12	Change Cipher Spec, Encrypted Handshake Message
59	2024-01-25 12:31:38.328702336...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=954 Ack=2591 Win=131008 Len=0 TSval=1676452265 TSecr=441496317
60	2024-01-25 12:31:38.151248214...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	562	12	Application Data
61	2024-01-25 12:31:38.257435452...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TCP	66	12	3128 -> 61484 [ACK] Seq=2591 Ack=1450 Win=64192 Len=0 TSval=441496387 TSecr=1676452265
64	2024-01-25 12:31:38.296760748...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLsv1.2	111	13	Application Data
73	2024-01-25 12:31:38.411911657...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	13	443 -> 24953 [ACK] Seq=3618 Ack=365 Win=67072 Len=0 TSval=1727281896 TSecr=2549354298
74	2024-01-25 12:31:38.340012513...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TLsv1.2	640	13	Application Data, Application Data
78	2024-01-25 12:31:39.283208060...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	13	443 -> 24953 [ACK] Seq=3618 Ack=939 Win=68096 Len=0 TSval=1727282019 TSecr=2549354468
79	2024-01-25 12:31:39.155943076...	93.184.216.34	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	1146	13	Application Data, Application Data
80	2024-01-25 12:31:39.305106563...	10.48.48.165	VMware_8d:9a:f4	93.184.216.34	Cisco_9d:b9:ff	TCP	66	13	24953 -> 443 [ACK] Seq=939 Ack=4698 Win=11456 Len=0 TSval=2549354588 TSecr=1727282020
82	2024-01-25 12:31:39.165086323...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	112	12	Application Data
83	2024-01-25 12:31:39.342008...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1450 Ack=2637 Win=131008 Len=0 TSval=1676452764 TSecr=44149680
84	2024-01-25 12:31:39.200484740...	10.48.48.165	VMware_8d:9a:f4	10.61.70.23	Cisco_9d:b9:ff	TLsv1.2	1209	12	Application Data, Application Data
85	2024-01-25 12:31:39.128618294...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TCP	66	12	61484 -> 3128 [ACK] Seq=1450 Ack=3700 Win=129920 Len=0 TSval=1676452838 TSecr=44149680
86	2024-01-25 12:31:39.092047...	10.61.70.23	Cisco_9d:b9:ff	10.48.48.165	VMware_8d:9a:f4	TLsv1.2	497	12	Application Data

Bild - Vollständiges explizites HTTPS ohne Cache



Hinweis: Jeder Datenverkehrsstrom zeichnet sich durch eine andere Farbe aus. Der Datenverkehrsfluss vom Client zum SWA ist eine Farbe, der Datenverkehrsfluss vom SWA zum Webserver eine andere.

Time	10.61.70.23	10.48.48.165	93.184.216.34	Comment
2024-01-25 12:31:37.3181686448 nanoseconds)	61484	61484 → 3128 [SYN] Seq=0 Win=65535 L...	3128	TCP: 61484 → 3128 [SYN] Seq=0 Win=65535 ...
2024-01-25 12:31:37.3300153152 nanoseconds)	61484	3128 → 61484 [SYN, ACK] Seq=0 Ack=1 ...	3128	TCP: 3128 → 61484 [SYN, ACK] Seq=0 Ack=1 ...
2024-01-25 12:31:37.3702977600 nanoseconds)	61484	61484 → 3128 [ACK] Seq=1 Ack=1 Win=13 ...	3128	TCP: 61484 → 3128 [ACK] Seq=1 Ack=1 Win=13 ...
2024-01-25 12:31:37.383167	61484	CONNECT example.com:443 HTTP/1.1	3128	HTTP: CONNECT example.com:443 HTTP/1.1
2024-01-25 12:31:37.3249466192 nanoseconds)	61484	3128 → 61484 [ACK] Seq=1 Ack=212 Win...	3128	TCP: 3128 → 61484 [ACK] Seq=1 Ack=212 Win...
2024-01-25 12:31:37.383901		24953 → 443 [SYN] Seq=0 Win=12288 Len...	443	TCP: 24953 → 443 [SYN] Seq=0 Win=12288 Len...
2024-01-25 12:31:38.006918		443 → 24953 [SYN, ACK] Seq=0 Ack=1 Win...	443	TCP: 443 → 24953 [SYN, ACK] Seq=0 Ack=1 Win...
2024-01-25 12:31:38.893381		24953 → 443 [ACK] Seq=1 Ack=1 Win=12...	443	TCP: 24953 → 443 [ACK] Seq=1 Ack=1 Win=12...
2024-01-25 12:31:38.731815	61484	HTTP/1.1 200 Connection established	3128	HTTP: HTTP/1.1 200 Connection established
2024-01-25 12:31:38.3088775616 nanoseconds)	61484	61484 → 3128 [ACK] Seq=212 Ack=40 Win...	3128	TCP: 61484 → 3128 [ACK] Seq=212 Ack=40 Win...
2024-01-25 12:31:38.3223471664 nanoseconds)	61484	Client Hello (SNI=example.com)	3128	TLSv1.2: Client Hello (SNI=example.com)
2024-01-25 12:31:38.1820724752 nanoseconds)	61484	3128 → 61484 [ACK] Seq=40 Ack=861 Win...	3128	TCP: 3128 → 61484 [ACK] Seq=40 Ack=861 Win...
2024-01-25 12:31:38.350314		Client Hello (SNI=example.com)	443	TLSv1.2: Client Hello (SNI=example.com)
2024-01-25 12:31:38.1465354064 nanoseconds)		443 → 24953 [ACK] Seq=1 Ack=194 Win...	443	TCP: 443 → 24953 [ACK] Seq=1 Ack=194 Win...
2024-01-25 12:31:38.2470315936 nanoseconds)		Server Hello	443	TLSv1.2: Server Hello
2024-01-25 12:31:38.2733499712 nanoseconds)		24953 → 443 [ACK] Seq=194 Ack=1369 ...	443	TCP: 24953 → 443 [ACK] Seq=194 Ack=1369 ...
2024-01-25 12:31:38.1414890096 nanoseconds)		443 → 24953 [PSH, ACK] Seq=1369 Ack...	443	TCP: 443 → 24953 [PSH, ACK] Seq=1369 Ack...
2024-01-25 12:31:38.1786810448 nanoseconds)		24953 → 443 [ACK] Seq=194 Ack=2737 ...	443	TCP: 24953 → 443 [ACK] Seq=194 Ack=2737 ...
2024-01-25 12:31:38.345520		Certificate, Server Key Exchange, Ser...	443	TLSv1.2: Certificate, Server Key Exchange, Ser...
2024-01-25 12:31:38.1610403440 nanoseconds)		24953 → 443 [ACK] Seq=194 Ack=3567 ...	443	TCP: 24953 → 443 [ACK] Seq=194 Ack=3567 ...
2024-01-25 12:31:38.062391		Client Key Exchange, Change Cipher Spec...	443	TLSv1.2: Client Key Exchange, Change Cipher ...
2024-01-25 12:31:38.4140285008 nanoseconds)		Change Cipher Spec, Encrypted Handshak...	443	TLSv1.2: Change Cipher Spec, Encrypted Hand...
2024-01-25 12:31:38.1095737424 nanoseconds)		24953 → 443 [ACK] Seq=320 Ack=3618 ...	443	TCP: 24953 → 443 [ACK] Seq=320 Ack=3618 ...
2024-01-25 12:31:38.2820976608 nanoseconds)	61484	Server Hello	3128	TLSv1.2: Server Hello
2024-01-25 12:31:38.1534298672 nanoseconds)	61484	Certificate	3128	TLSv1.2: Certificate
2024-01-25 12:31:38.965425	61484	Server Key Exchange, Server Hello Done	3128	TLSv1.2: Server Key Exchange, Server Hello D...
2024-01-25 12:31:38.824826	61484	61484 → 3128 [ACK] Seq=861 Ack=1228 ...	3128	TCP: 61484 → 3128 [ACK] Seq=861 Ack=1228 ...
2024-01-25 12:31:38.3446619136 nanoseconds)	61484	61484 → 3128 [ACK] Seq=861 Ack=2540 ...	3128	TCP: 61484 → 3128 [ACK] Seq=861 Ack=2540 ...
2024-01-25 12:31:38.1738329504 nanoseconds)	61484	Client Key Exchange, Change Cipher Spec...	3128	TLSv1.2: Client Key Exchange, Change Cipher ...
2024-01-25 12:31:38.4228567872 nanoseconds)	61484	3128 → 61484 [ACK] Seq=2540 Ack=954 ...	3128	TCP: 3128 → 61484 [ACK] Seq=2540 Ack=954 ...
2024-01-25 12:31:38.2445141472 nanoseconds)	61484	Change Cipher Spec, Encrypted Handshak...	3128	TLSv1.2: Change Cipher Spec, Encrypted Hand...
2024-01-25 12:31:38.3287023360 nanoseconds)	61484	61484 → 3128 [ACK] Seq=954 Ack=2591 ...	3128	TCP: 61484 → 3128 [ACK] Seq=954 Ack=2591 ...

Image - HTTPS-Fluss - explizit - kein Cache

Beispiel für AccessLogs:

```
1706174571.215 582 10.61.70.23 TCP_MISS_SSL/200 39 CONNECT tunnel://www.example.com:443/ - DIRECT/www.e
1706174571.486 270 10.61.70.23 TCP_MISS_SSL/200 1106 GET https://www.example.com:443/ - DIRECT/www.exam
```



Hinweis: Wie Sie bei einer transparenten Bereitstellung für HTTPS-Datenverkehr sehen können, gibt es in AccessLogs zwei Zeilen. Die erste Zeile lautet, wenn der Datenverkehr verschlüsselt ist, und Sie können CONNECT sehen, und die URL des Webservers beginnt mit tunnel://. Wenn die Entschlüsselung in SWA aktiviert ist, enthält die zweite Zeile GET, und die gesamte URL beginnt mit HTTPS, d. h. der Datenverkehr wurde entschlüsselt.

HTTPS-Passthrough-Datenverkehr

Wenn Sie die SWA so konfiguriert haben, dass der Datenverkehr weitergeleitet wird, lautet der Gesamtfluss wie folgt:

Time	10.61.70.23	10.48.48.165	93.184.216.34	Comment
2024-01-25 13:21:42.706645	60250	60250 → 3128 [SYN] Seq=0 Win=65535 Len=0 MSS=1260 WS=64 TSval=341363	3128	TCP: 60250 → 3128 [SYN] Seq=0 Win=65535 ...
2024-01-25 13:21:42.2460867504 (nanoseconds)	60250	3128 → 60250 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SA	3128	TCP: 3128 → 60250 [SYN, ACK] Seq=0 Ack=1 ...
2024-01-25 13:21:42.1279136912 (nanoseconds)	60250	60250 → 3128 [ACK] Seq=1 Ack=1 Win=132288 Len=0 TSval=341363763 TSecr=1	3128	TCP: 60250 → 3128 [ACK] Seq=1 Ack=1 Win=1...
2024-01-25 13:21:42.4235993424 (nanoseconds)	60250	CONNECT example.com:443 HTTP/1.1	3128	HTTP: CONNECT example.com:443 HTTP/1.1
2024-01-25 13:21:42.2468178944 (nanoseconds)	60250	3128 → 60250 [ACK] Seq=1 Ack=212 Win=65344 Len=0 TSval=1253711229 TSecr=1	3128	TCP: 3128 → 60250 [ACK] Seq=1 Ack=212 Win...
2024-01-25 13:21:42.1692445712 (nanoseconds)			17517	17517 → 443 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSv...
2024-01-25 13:21:42.1675493712 (nanoseconds)			17517	443 → 17517 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM...
2024-01-25 13:21:42.402773			17517	17517 → 443 [ACK] Seq=1 Ack=1 Win=12...
2024-01-25 13:21:42.3955843776 (nanoseconds)	60250	HTTP/1.1 200 Connection established	3128	HTTP: HTTP/1.1 200 Connection established
2024-01-25 13:21:42.044443	60250	60250 → 3128 [ACK] Seq=212 Ack=40 Win=132224 Len=0 TSval=341363960 TSecr=1	3128	TCP: 60250 → 3128 [ACK] Seq=212 Ack=40 W...
2024-01-25 13:21:42.2651980528 (nanoseconds)	60250	Client Hello (SNI=example.com)	3128	TLV.1.3: Client Hello (SNI=example.com)
2024-01-25 13:21:42.1640450432 (nanoseconds)	60250	3128 → 60250 [ACK] Seq=40 Ack=861 Win=64704 Len=0 TSval=1253711429 TSecr=1	3128	TCP: 3128 → 60250 [ACK] Seq=40 Ack=861 W...
2024-01-25 13:21:42.2261550016 (nanoseconds)			17517	Client Hello (SNI=example.com)
2024-01-25 13:21:42.2572160048 (nanoseconds)			17517	443 → 17517 [ACK] Seq=1 Ack=650 Win=...
2024-01-25 13:21:42.310233			17517	Server Hello, Change Cipher Spec, Application Data
2024-01-25 13:21:42.1377394032 (nanoseconds)			17517	17517 → 443 [ACK] Seq=650 Ack=1369 W...
2024-01-25 13:21:42.1401624816 (nanoseconds)			17517	443 → 17517 [PSH, ACK] Seq=1369 Ack=650 Win=67072 Len=1368 TSval=179516...
2024-01-25 13:21:42.2565014960 (nanoseconds)	60250	Server Hello, Change Cipher Spec, Application Data	3128	TLV.1.3: Server Hello, Change Cipher Spec, Ap...
2024-01-25 13:21:42.1431156304 (nanoseconds)			17517	17517 → 443 [ACK] Seq=650 Ack=2737 Win=11072 Len=0 TSval=900013138 TSecr=1
2024-01-25 13:21:42.2106897872 (nanoseconds)	60250	3128 → 60250 [PSH, ACK] Seq=1228 Ack=861 Win=64704 Len=180 TSval=125371	3128	TCP: 3128 → 60250 [PSH, ACK] Seq=1228 Ack...
2024-01-25 13:21:42.3887370384 (nanoseconds)	60250	3128 → 60250 [ACK] Seq=1408 Ack=861 Win=64704 Len=188 TSval=125371160...	3128	TCP: 3128 → 60250 [ACK] Seq=1408 Ack=861...
2024-01-25 13:21:42.3839993744 (nanoseconds)	60250	3128 → 60250 [PSH, ACK] Seq=2596 Ack=861 Win=64704 Len=180 TSval=12537...	3128	TCP: 3128 → 60250 [PSH, ACK] Seq=2596 Ac...
2024-01-25 13:21:42.1001611472 (nanoseconds)			17517	Application Data, Application Data
2024-01-25 13:21:42.3850714352 (nanoseconds)			17517	17517 → 443 [ACK] Seq=650 Ack=4105 Win=11072 Len=0 TSval=900013138 TSecr=1
2024-01-25 13:21:42.542333	60250	Application Data	3128	TLV.1.3: Application Data
2024-01-25 13:21:42.2351706320 (nanoseconds)	60250	Application Data	3128	TLV.1.3: Application Data
2024-01-25 13:21:42.4080650144 (nanoseconds)			17517	Application Data
2024-01-25 13:21:42.3133660336 (nanoseconds)			17517	17517 → 443 [ACK] Seq=650 Ack=4171 Win=12416 Len=0 TSval=900013138 TSecr=1
2024-01-25 13:21:42.3354894224 (nanoseconds)	60250	Application Data	3128	TLV.1.3: Application Data
2024-01-25 13:21:42.400703	60250	60250 → 3128 [ACK] Seq=861 Ack=1228 Win=131008 Len=0 TSval=341364213...	3128	TCP: 60250 → 3128 [ACK] Seq=861 Ack=1228 ...
2024-01-25 13:21:42.367120	60250	60250 → 3128 [ACK] Seq=861 Ack=4210 Win=128064 Len=0 TSval=341364213...	3128	TCP: 60250 → 3128 [ACK] Seq=861 Ack=4210...
2024-01-25 13:21:42.2112887360 (nanoseconds)	 [TCP Window Update] 60250 → 3128 [ACK] Seq=861 Ack=4210 Win=131072 Len=0		TCP: [TCP Window Update] 60250 → 3128 [AC...

Image - HTTPS-Passthrough - Explizit - Fluss

Beispiel für Client Hello von SWA auf Webserver:

- Transport Layer Security
 - TLV.1.3 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 644
 - Handshake Protocol: Client Hello
 - Handshake Type: Client Hello (1)
 - Length: 640
 - Version: TLS 1.2 (0x0303)
 - Random: 2c545a566b5b3f338dc9dbd80ea91ad61035c786954ced219e266ff0b92b9c1
 - Session ID Length: 32
 - Session ID: 86da348af5508fc24f18f3cbd9829c7282b77e0499e5d2f38466ccbd66821e2
 - Cipher Suites Length: 34
 - Cipher Suites (17 suites)
 - Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
 - Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
 - Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xc03a)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xc038)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
 - Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
 - Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
 - Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
 - Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
 - Compression Methods Length: 1
 - Compression Methods (1 method)
 - Extensions Length: 533
 - Extension: server_name (len=16) name=example.com
 - Type: server_name (0)
 - Length: 16
 - Server Name Indication extension
 - Server Name list length: 14
 - Server Name Type: host_name (0)
 - Server Name length: 11
 - Server Name: example.com
 - Extension: extended_master_secret (len=0)
 - Extension: renegotiation_info (len=1)
 - Extension: supported_groups (len=14)
 - Extension: ec_point_formats (len=2)

Bild - HTTPS-Passthrough - Explizit - SWA an Webserver - Client hello

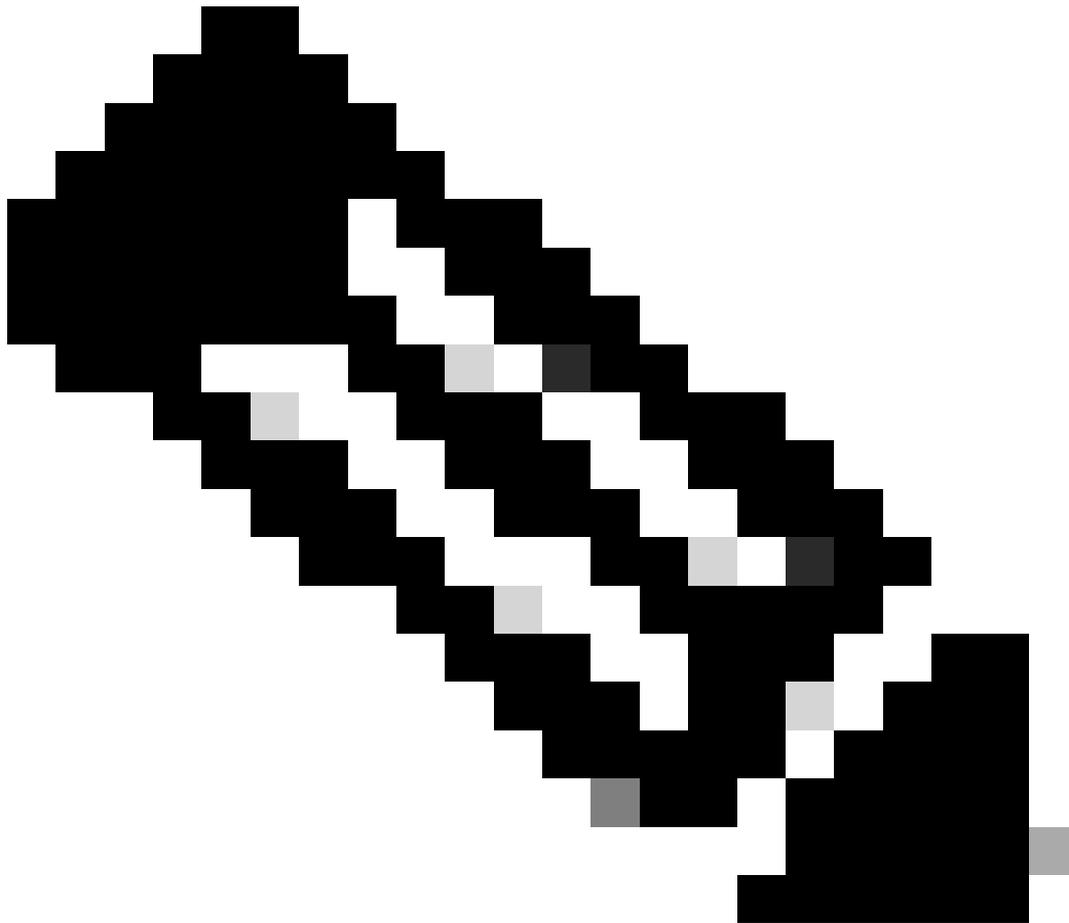
Dies entspricht dem "Client Hello"-Vorgang vom Client zur SWA:

- ▼ Transport Layer Security
 - ▼ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 644
 - ▼ Handshake Protocol: Client Hello
 - Handshake Type: Client Hello (1)
 - Length: 640
 - Version: TLS 1.2 (0x0303)
 - Random: 2c545a566b5b3f338dc9dbd80ea91ad61035c786954ced2191e266ff0b92b9c1
 - Session ID Length: 32
 - Session ID: 86da348af5508fc24f18f3cbd9829c7282b77e0499e5d2f38466cccbd66821e2
 - Cipher Suites Length: 34
 - ▼ Cipher Suites (17 suites)
 - Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
 - Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
 - Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xc030)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xc031)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc032)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc033)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc034)
 - Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc035)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc036)
 - Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc037)
 - Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0xc038)
 - Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0xc039)
 - Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0xc03a)
 - Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0xc03b)
 - Compression Methods Length: 1
 - > Compression Methods (1 method)
 - Extensions Length: 533
 - ▼ Extension: server_name (len=16) name=example.com
 - Type: server_name (0)
 - Length: 16
 - ▼ Server Name Indication extension
 - Server Name list length: 14
 - Server Name Type: host_name (0)
 - Server Name length: 11
 - Server Name: example.com
 - ▼ Extension: extended_master_secret (len=0)
 - Type: extended_master_secret (23)
 - Length: 0
 - ▼ Extension: renegotiation_info (len=1)

Bild - HTTPS-Passthrough - Explizit - Client an SWA - Client hello

Hier ein Beispiel für ein Accesslog:

1706185288.920 53395 10.61.70.23 TCP_MISS/200 6549 CONNECT tunnel://www.example.com:443/ - DIRECT/www.e



Hinweis: Wie Sie sehen können, ist es nur eine Zeile und die Aktion ist PASSTHRU.

Transparente Bereitstellung

HTTP-Datenverkehr bei transparenter Bereitstellung ohne Authentifizierung

Client und SWA

Netzwerkverkehr wird zwischen der IP-Adresse des Clients und der IP-Adresse des Webserver übertragen.

Der Datenverkehr vom Client ist für TCP-Port 80 bestimmt (nicht für den Proxy-Port)

- TCP-Handshake
- HTTP Get from Client (Ziel-IP = Webserver, Ziel-Port = 80)
- HTTP-Antwort vom Proxy (Quell-IP = Webserver)

- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Length	Stream	Info
7	2023-12-11 19:13:47.372486256	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	66	0	54468 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
-	2023-12-11 19:13:47.243585552	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	66	0	80 → 54468 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM
-	2023-12-11 19:13:47.267161713	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 → 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0
-	2023-12-11 19:13:47.388984368	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	HTTP	128	0	GET / HTTP/1.1
-	2023-12-11 19:13:47.624692	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 → 54468 [ACK] Seq=1 Ack=75 Win=65472 Len=0
-	2023-12-11 19:13:47.285645694	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	1514	0	80 → 54468 [ACK] Seq=1 Ack=75 Win=65472 Len=1460 [TCP segment of a reassembled PDU]
-	2023-12-11 19:13:47.237549915	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	HTTP	381	0	HTTP/1.1 200 OK (text/html)
-	2023-12-11 19:13:47.2669807	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 → 80 [ACK] Seq=75 Ack=1788 Win=262656 Len=0
-	2023-12-11 19:13:47.353942364	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 → 80 [FIN, ACK] Seq=75 Ack=1788 Win=262656 Len=0
-	2023-12-11 19:13:47.266665884	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 → 54468 [ACK] Seq=1788 Ack=76 Win=65472 Len=0
-	2023-12-11 19:13:47.111822518	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 → 54468 [FIN, ACK] Seq=1788 Ack=76 Win=65472 Len=0
-	2023-12-11 19:13:47.168465673	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 → 80 [ACK] Seq=76 Ack=1789 Win=262656 Len=0

Image - Client zu Proxy - HTTP - Transparent - Keine Authentifizierung

Hier ist ein Beispiel für HTTP Get from Client.

```

> Frame 11: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits)
> Ethernet II, Src: Cisco_76:fb:16 (70:70:8b:76:fb:16), Dst: Cisco_56:5f:44 (68:bd:ab:56:5f:44)
> Internet Protocol Version 4, Src: 10.201.189.180, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 65132, Dst Port: 80, Seq: 1, Ack: 1, Len: 177
< Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Connection: keep-alive\r\n
    Host: example.com\r\n
    User-Agent: curl/8.4.0\r\n
    Accept: */*\r\n
    X-IMForwards: 20\r\n
    Via: 1.1 wsa695948022.calolab.com:80 (Cisco-WSA/15.0.0-355)\r\n
    \r\n
    [Full request URI: http://example.com/]
    [HTTP request 1/1]
    [Response in frame: 15]
  
```

Image - Client zu Proxy - HTTP - Transparent - Keine Auth - Client HTTP Get

SWA und Webserver

Der Netzwerkverkehr erfolgt zwischen der IP-Adresse des Proxys und der IP-Adresse des Webserver.

Der Datenverkehr von SWA ist für TCP-Port 80 bestimmt (nicht für den Proxy-Port)

- TCP-Handshake
- HTTP Get from Proxy (Ziel-IP = Webserver, Ziel-Port = 80)
- HTTP-Antwort vom Webserver (Quell-IP = Proxyserver)
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Length	Stream	Info
8	2023-12-11 19:13:47.260946116	10.201.189.180	Cisco_56:5f:44	93.184.216.34	Cisco_56:5f:44	TCP	74	1	65132 → 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSval=1559577035 TSecr=0
9	2023-12-11 19:13:47.273148633	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	74	1	80 → 65132 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM TSval=6873333 TSecr=6873463
10	2023-12-11 19:13:47.285008827	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 → 80 [ACK] Seq=1 Ack=1 Win=13184 Len=0 TSval=1559577165 TSecr=6873333
11	2023-12-11 19:13:47.307381585	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	HTTP	243	1	GET / HTTP/1.1
12	2023-12-11 19:13:47.118451681	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 → 65132 [ACK] Seq=1 Ack=178 Win=66368 Len=0 TSval=6873333 TSecr=1559577035
13	2023-12-11 19:13:47.209157872	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	1514	1	80 → 65132 [ACK] Seq=1 Ack=178 Win=66368 Len=1448 TSval=6873463 TSecr=1559577035 [TCP segment of a reassembled PDU]
14	2023-12-11 19:13:47.637333	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 → 80 [ACK] Seq=178 Ack=1449 Win=11776 Len=0 TSval=1559577165 TSecr=6873463
15	2023-12-11 19:13:47.276272012	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	HTTP	349	1	HTTP/1.1 200 OK (text/html)
16	2023-12-11 19:13:47.249979843	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 → 80 [ACK] Seq=178 Ack=1732 Win=11520 Len=0 TSval=1559577165 TSecr=6873463
17	2023-12-11 19:14:12.270488529	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 → 80 [FIN, ACK] Seq=178 Ack=1732 Win=13184 Len=0 TSval=1559602015 TSecr=6873463
18	2023-12-11 19:14:12.236807	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 → 65132 [ACK] Seq=1732 Ack=179 Win=66368 Len=0 TSval=6898313 TSecr=1559602015
19	2023-12-11 19:14:12.215970816	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 → 65132 [FIN, ACK] Seq=1732 Ack=179 Win=66368 Len=0 TSval=6898313 TSecr=1559602015
20	2023-12-11 19:14:12.218303318	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 → 80 [ACK] Seq=179 Ack=1733 Win=13120 Len=0 TSval=1559602015 TSecr=6898313

Image- Proxy und Webserver - HTTP - Transparent - Keine Authentifizierung

Nachfolgend finden Sie ein Beispiel für den HTTP-Abruf vom Proxy.

```

> Frame 20: 128 bytes on wire (1024 bits), 128 bytes captured (1024 bits)
> Ethernet II, Src: Cisco_c9:c0:7f (74:88:bb:c9:c0:7f), Dst: Cisco_76:fb:15 (70:70:8b:76:fb:15)
> Internet Protocol Version 4, Src: 192.168.1.10, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 54468, Dst Port: 80, Seq: 1, Ack: 1, Len: 74
< Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Host: example.com\r\n
    User-Agent: curl/8.4.0\r\n
    Accept: */*\r\n
    \r\n
    [Full request URI: http://example.com/]
    [HTTP request 1/1]
    [Response in frame: 23]

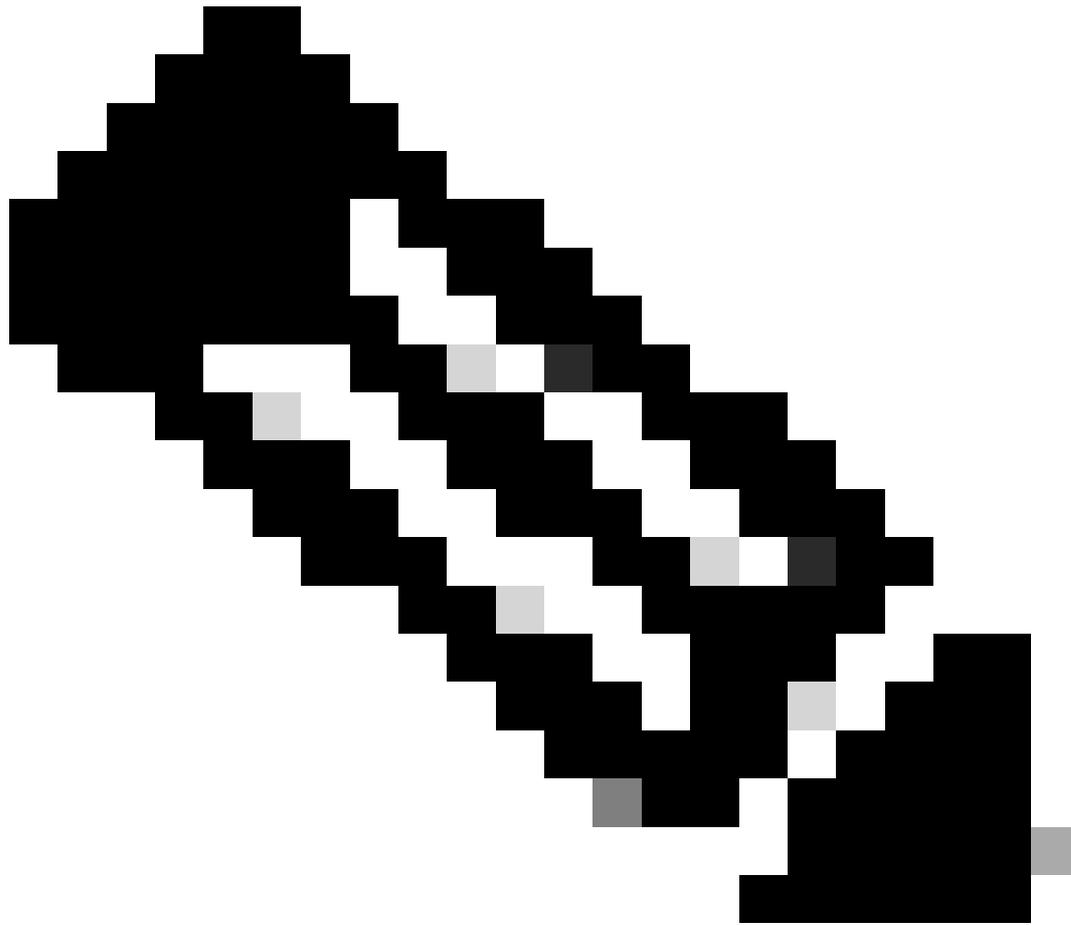
```

Image- Proxy an Webservice - HTTP - Transparent - Keine Auth - Proxy HTTP Get

Dieser Wert stellt den gesamten Datenverkehrsfluss vom Client zum SWA, dann zum Webserver und schließlich zurück zum Client dar.

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Len	stream	Info
7	2023-12-11 19:13:47.372486256	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	66	0	54468 -> 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
8	2023-12-11 19:13:47.260946116	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	74	1	65132 -> 80 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSval=1559577035 TSecr=0
9	2023-12-11 19:13:47.273148633	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	74	1	80 -> 65132 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM TSval=6873333 TSecr=
10	2023-12-11 19:13:47.285008027	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 -> 80 [ACK] Seq=1 Ack=1 Win=13184 Len=0 TSval=1559577035 TSecr=6873333
11	2023-12-11 19:13:47.307381585	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	HTTP	243	1	GET / HTTP/1.1
12	2023-12-11 19:13:47.118451681	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 -> 65132 [ACK] Seq=1 Ack=178 Win=66368 Len=0 TSval=6873333 TSecr=1559577035
13	2023-12-11 19:13:47.029167872	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	1514	1	80 -> 65132 [ACK] Seq=1 Ack=178 Win=66368 Len=1448 TSval=6873463 TSecr=1559577035 [TCP segment
14	2023-12-11 19:13:47.637333	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 -> 80 [ACK] Seq=178 Ack=1449 Win=11776 Len=0 TSval=1559577165 TSecr=6873463
15	2023-12-11 19:13:47.276272012	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	HTTP	349	1	HTTP/1.1 200 OK (text/html)
16	2023-12-11 19:13:47.249979843	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 -> 80 [ACK] Seq=178 Ack=1732 Win=11520 Len=0 TSval=1559577165 TSecr=6873463
18	2023-12-11 19:13:47.243585552	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	66	0	80 -> 54468 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM
19	2023-12-11 19:13:47.267161713	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 -> 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0
20	2023-12-11 19:13:47.388984368	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	HTTP	128	0	GET / HTTP/1.1
21	2023-12-11 19:13:47.624692	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 -> 54468 [ACK] Seq=1 Ack=75 Win=65472 Len=0
22	2023-12-11 19:13:47.285645694	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	1514	0	80 -> 54468 [ACK] Seq=1 Ack=75 Win=65472 Len=1460 [TCP segment of a reassembled PDU]
23	2023-12-11 19:13:47.237549915	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	HTTP	381	0	HTTP/1.1 200 OK (text/html)
24	2023-12-11 19:13:47.266907	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 -> 80 [ACK] Seq=75 Ack=1788 Win=262656 Len=0
25	2023-12-11 19:13:47.355942364	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 -> 80 [FIN, ACK] Seq=75 Ack=1788 Win=262656 Len=0
26	2023-12-11 19:13:47.266665804	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 -> 54468 [ACK] Seq=1788 Ack=76 Win=5472 Len=0
27	2023-12-11 19:13:47.111822518	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	0	80 -> 54468 [FIN, ACK] Seq=1788 Ack=76 Win=5472 Len=0
28	2023-12-11 19:13:47.168465673	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	0	54468 -> 80 [ACK] Seq=76 Ack=1789 Win=262656 Len=0
1.	2023-12-11 19:14:12.270488529	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 -> 80 [FIN, ACK] Seq=178 Ack=1732 Win=13184 Len=0 TSval=1559602015 TSecr=6873463
1.	2023-12-11 19:14:12.236807	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 -> 65132 [ACK] Seq=1732 Ack=179 Win=66368 Len=0 TSval=6898313 TSecr=1559602015
1.	2023-12-11 19:14:12.215970816	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80 -> 65132 [FIN, ACK] Seq=1732 Ack=179 Win=66368 Len=0 TSval=6898313 TSecr=1559602015
1.	2023-12-11 19:14:12.218303318	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	65132 -> 80 [ACK] Seq=179 Ack=1733 Win=13120 Len=0 TSval=1559602015 TSecr=6898313

Bild - Gesamtverkehr - HTTP - Transparent - Keine Authentifizierung



Hinweis: Jeder Datenverkehrsstrom zeichnet sich durch eine andere Farbe aus. Der Datenverkehrsfluss vom Client zum SWA ist eine Farbe, der Datenverkehrsfluss vom SWA zum Webserver eine andere.



Bild - WCCP-HTTP-Fluss

Beispiel für AccessLogs:

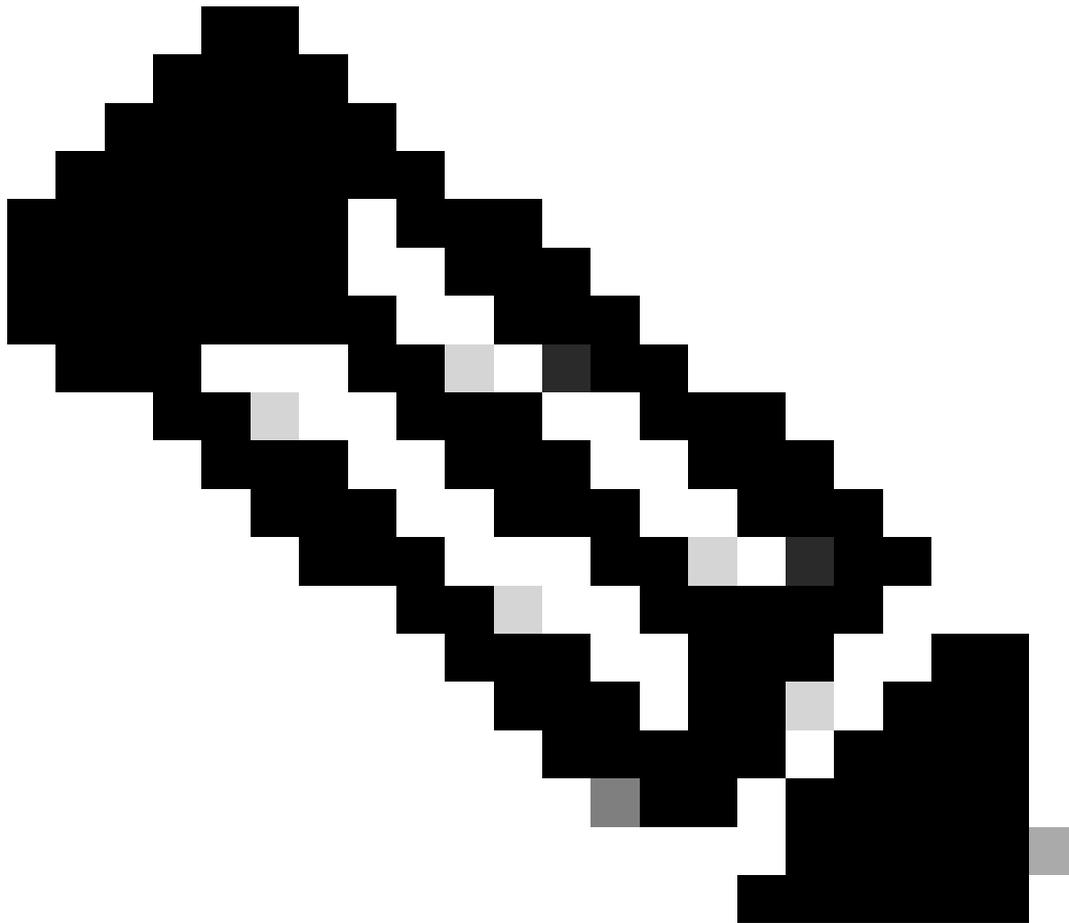
1702318427.181 124 192.168.1.10 TCP_MISS/200 1787 GET http://www.example.com/ - DIRECT/www.example.com

Datenverkehr mit zwischengespeicherten Daten

Dieser Wert stellt den gesamten Datenverkehrsfluss vom Client zum SWA dar, wenn sich die Daten im SWA-Cache befinden.

Line	Time	Source IP	Source Port	Source Device	Destination IP	Destination Port	Destination Device	Protocol	Length	Flags	Seq	Win	MSS	WS	SACK	PERM	TSeq	TSval	TSecr	Comment				
9	2023-12-11 19:19:49.	111544768..		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	74	1	13586	-	80	[SYN]	Seq=0	Win=12288	Len=0	MSS=1460	WS=64	SACK_PERM	TSeq=0	TSval=3178850246	TSecr=0	
11	2023-12-11 19:19:49.	259539926..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	66	2	54487	-	80	[SYN]	Seq=0	Win=64240	Len=0	MSS=1460	WS=256	SACK_PERM				
12	2023-12-11 19:19:49.	254858128..		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	66	2	80	-	54487	[SYN, ACK]	Seq=0	Ack=1	Win=65535	Len=0	MSS=1460	WS=64	SACK_PERM			
13	2023-12-11 19:19:49.	272497027..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	2	54487	-	80	[ACK]	Seq=1	Ack=1	Win=262656	Len=0						
14	2023-12-11 19:19:49.	178847280..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	HTTP	128	2	GET / HTTP/1.1													
15	2023-12-11 19:19:49.	184967324..		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	2	80	-	54487	[ACK]	Seq=1	Ack=75	Win=65472	Len=0						
16	2023-12-11 19:19:49.	6565285		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	1514	2	80	-	54487	[ACK]	Seq=1	Ack=75	Win=65472	Len=1460	[TCP segment of a reassembled PDU]					
17	2023-12-11 19:19:49.	425926200..		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	HTTP	381	2	HTTP/1.1 200 OK (text/html)													
18	2023-12-11 19:19:49.	278838524..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	2	54487	-	80	[ACK]	Seq=75	Ack=1788	Win=262656	Len=0						
19	2023-12-11 19:19:49.	391018345..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	2	54487	-	80	[FIN, ACK]	Seq=75	Ack=1788	Win=262656	Len=0						
20	2023-12-11 19:19:49.	394258659..		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	2	80	-	54487	[ACK]	Seq=1788	Ack=76	Win=65472	Len=0						
21	2023-12-11 19:19:49.	910090		93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	2	80	-	54487	[FIN, ACK]	Seq=1788	Ack=76	Win=65472	Len=0						
22	2023-12-11 19:19:49.	179047075..		192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	2	54487	-	80	[ACK]	Seq=76	Ack=1789	Win=262656	Len=0						
23	2023-12-11 19:19:49.	372291046..		93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	74	1	80	-	13586	[SYN, ACK]	Seq=0	Ack=1	Win=65535	Len=0	MSS=1460	WS=64	SACK_PERM	TSeq=0	TSval=4080954250	TSecr=0
24	2023-12-11 19:19:49.	309178142..		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	13586	-	80	[ACK]	Seq=1	Ack=1	Win=13184	Len=0	TSval=3178850246	TSecr=4080954250				
25	2023-12-11 19:19:49.	226286489..		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	HTTP	293	1	GET / HTTP/1.1													
26	2023-12-11 19:19:49.	207193169..		93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80	-	13586	[ACK]	Seq=1	Ack=228	Win=66368	Len=0	TSval=4080954250	TSecr=3178850246				
27	2023-12-11 19:19:49.	2279948003..		93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	HTTP	489	1	HTTP/1.1 304 Not Modified													
28	2023-12-11 19:19:49.	1336540662..		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	13586	-	80	[ACK]	Seq=228	Ack=424	Win=12800	Len=0	TSval=3178850356	TSecr=4080954361				
29	2023-12-11 19:19:49.	352537		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	13586	-	80	[FIN, ACK]	Seq=228	Ack=424	Win=13184	Len=0	TSval=3178850356	TSecr=4080954361				
30	2023-12-11 19:19:49.	194154916..		93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80	-	13586	[ACK]	Seq=424	Ack=229	Win=66368	Len=0	TSval=4080954361	TSecr=3178850356				
31	2023-12-11 19:19:49.	349158924..		93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	1	80	-	13586	[FIN, ACK]	Seq=424	Ack=229	Win=66368	Len=0	TSval=4080954361	TSecr=3178850356				
32	2023-12-11 19:19:49.	103444988..		10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	1	13586	-	80	[ACK]	Seq=229	Ack=425	Win=13120	Len=0	TSval=3178850356	TSecr=4080954361				

Bild - Zwischengespeichert - Gesamtverkehr - HTTP - Transparent - Keine Authentifizierung



Hinweis: Wie Sie sehen, gibt der Webserver die HTTP-Antwort 304: Cache not Modified (Cache nicht geändert) zurück. (in diesem Beispiel: Paketnummer 27)

Nachfolgend finden Sie ein Beispiel für die HTTP-Antwort 304.

```

> Frame 27: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)
> Ethernet II, Src: Cisco_56:5f:44 (68:bd:ab:56:5f:44), Dst: Cisco_76:fb:16 (70:70:8b:76:fb:16)
> Internet Protocol Version 4, Src: 93.184.216.34, Dst: 10.201.189.180
> Transmission Control Protocol, Src Port: 80, Dst Port: 13586, Seq: 1, Ack: 228, Len: 423
< Hypertext Transfer Protocol
  > HTTP/1.1 304 Not Modified\r\n
    Accept-Ranges: bytes\r\n
    Cache-Control: max-age=604800\r\n
    Date: Mon, 11 Dec 2023 18:22:17 GMT\r\n
    Etag: "3147526947"\r\n
    Expires: Mon, 18 Dec 2023 18:22:17 GMT\r\n
    Server: ECS (dce/26C6)\r\n
    Vary: Accept-Encoding\r\n
    X-Cache: HIT\r\n
    Last-Modified: Thu, 17 Oct 2019 07:18:26 GMT\r\n
    Age: 492653\r\n
    Via: 1.1 rtp1-lab-wsa-1.cisco.com:80 (Cisco-WSA/X), 1.1 proxy.rcdn.local:80 (Cisco-WSA/12.5.5-004)\r\n
    Connection: keep-alive\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.036615136 seconds]
    [Request in frame: 25]
    [Request URI: http://example.com/]

```

Bild - Zwischengespeichert - HTTP-Antwort 304 - HTTP - Transparent - Keine Authentifizierung

Beispiel für AccessLogs:

1702318789.560 105 192.168.1.10 TCP_REFRESH_HIT/200 1787 GET http://www.example.com/ - DIRECT/www.examp

HTTP-Datenverkehr bei transparenter Bereitstellung ohne Authentifizierung

Client und SWA

Netzwerkverkehr wird zwischen der IP-Adresse des Clients und der IP-Adresse des Webserverns übertragen.

Der Datenverkehr vom Client ist für TCP-Port 443 bestimmt (nicht für den Proxy-Port)

- TCP-Handshake
- TLS Handshake Client Hello - Server Hello - Serverschlüsselaustausch - Clientschlüsselaustausch
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Lengt	stream	Info
243	2023-12-11 19:36:24.416304924.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	66	14	54515 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
245	2023-12-11 19:36:24.139934096.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	66	14	443 → 54515 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM
246	2023-12-11 19:36:24.139934096.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	14	54515 → 443 [ACK] Seq=1 Ack=1 Win=262656 Len=0
247	2023-12-11 19:36:24.380754096.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TLSv1.	242	14	Client Hello (SNI=example.com)
248	2023-12-11 19:36:24.366528476.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [ACK] Seq=1 Ack=189 Win=65488 Len=0
256	2023-12-11 19:36:24.251614876.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	1514	14	Server Hello
257	2023-12-11 19:36:24.195519830.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	1043	14	Certificate, Server Key Exchange, Server Hello Done
258	2023-12-11 19:36:24.186747024.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	14	54515 → 443 [ACK] Seq=189 Ack=2450 Win=262656 Len=0
259	2023-12-11 19:36:24.299229398.	93.184.216.34	Cisco_c9:c0:7f	192.168.1.10	Cisco_76:fb:15	TLSv1.	147	14	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
260	2023-12-11 19:36:24.258163651.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [ACK] Seq=2450 Ack=282 Win=65344 Len=0
261	2023-12-11 19:36:24.299229398.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	105	14	Change Cipher Spec, Encrypted Handshake Message
262	2023-12-11 19:36:24.215995475.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TLSv1.	157	14	Application Data
263	2023-12-11 19:36:24.298152051.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [ACK] Seq=2501 Ack=385 Win=65280 Len=0
264	2023-12-11 19:36:25.529330	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	100	14	Application Data
265	2023-12-11 19:36:25.994499	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	1514	14	Application Data
266	2023-12-11 19:36:25.413207139.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	14	54515 → 443 [ACK] Seq=385 Ack=4007 Win=262656 Len=0
267	2023-12-11 19:36:25.201453091.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TLSv1.	311	14	Application Data
268	2023-12-11 19:36:25.181582688.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TLSv1.	85	14	Encrypted Alert
269	2023-12-11 19:36:25.404992054.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [ACK] Seq=4264 Ack=416 Win=65280 Len=0
278	2023-12-11 19:36:25.186927132.	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	14	54515 → 443 [FIN, ACK] Seq=416 Ack=4264 Win=262400 Len=0
271	2023-12-11 19:36:25.378433091.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [ACK] Seq=4264 Ack=417 Win=65280 Len=0
272	2023-12-11 19:36:25.342494763.	93.184.216.34	Cisco_76:fb:15	192.168.1.10	Cisco_c9:c0:7f	TCP	54	14	443 → 54515 [FIN, ACK] Seq=4264 Ack=417 Win=65280 Len=0
273	2023-12-11 19:36:25.794348	192.168.1.10	Cisco_c9:c0:7f	93.184.216.34	Cisco_76:fb:15	TCP	60	14	54515 → 443 [ACK] Seq=417 Ack=4265 Win=262400 Len=0

Image - Client zu Proxy - HTTPs - Transparent - Keine Authentifizierung

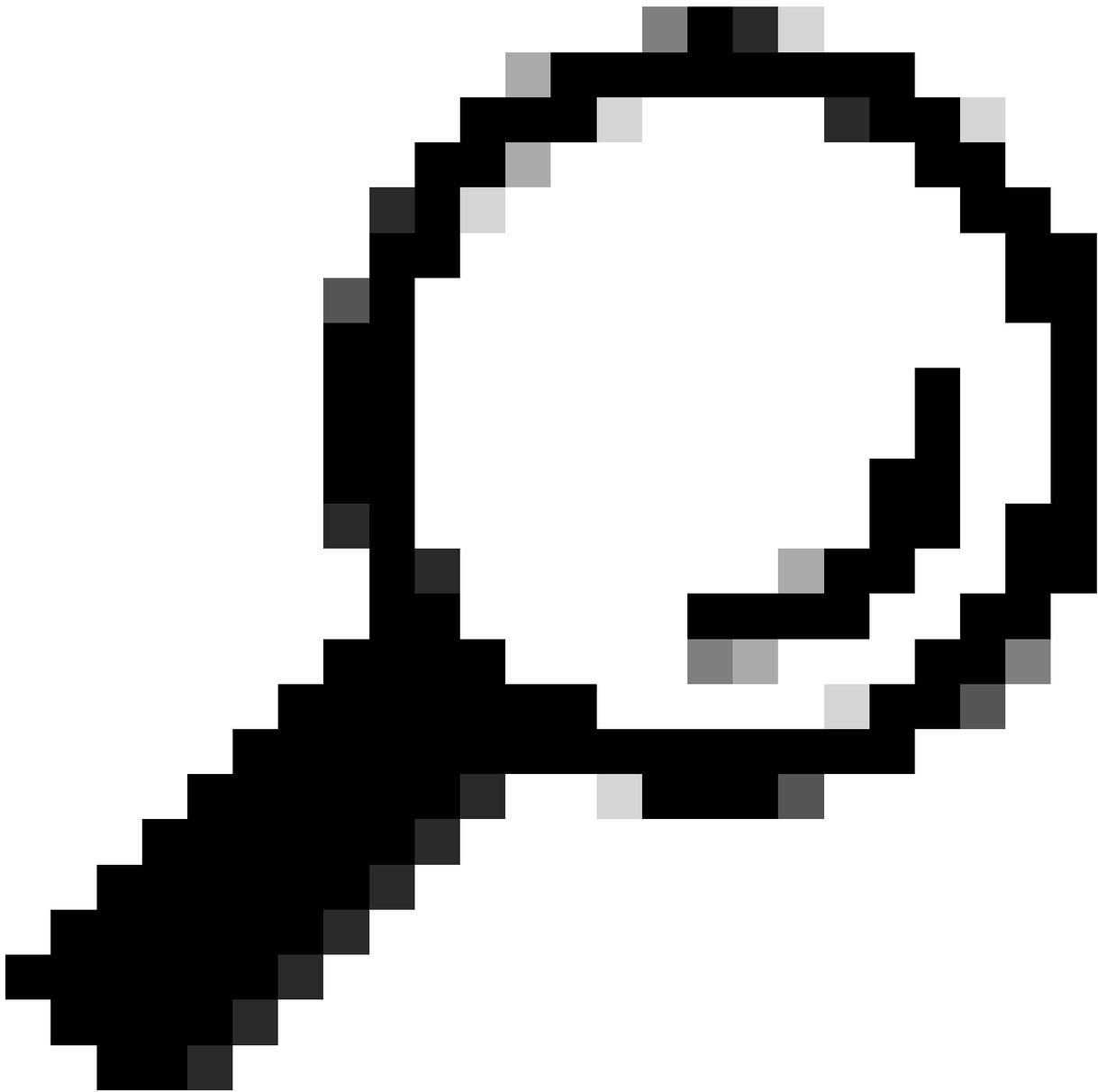
Hier sind Details von Client Hello von Client an SWA, wie Sie in der Server Name Indication (SNI) die URL des Webservers sehen können, die in diesem Beispiel www.example.com ist.

```

> Frame 247: 242 bytes on wire (1936 bits), 242 bytes captured (1936 bits)
> Ethernet II, Src: Cisco_c9:c0:7f (74:88:bb:c9:c0:7f), Dst: Cisco_76:fb:15 (70:70:8b:76:fb:15)
> Internet Protocol Version 4, Src: 192.168.1.10, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 54515, Dst Port: 443, Seq: 1, Ack: 1, Len: 188
< Transport Layer Security
  < TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 183
    < Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 179
      Version: TLS 1.2 (0x0303)
      Random: 657756ab224a3f64600e99172a8d38f86b689c7eb4bb121bf5408c96540a0f5d
      Session ID Length: 0
      Cipher Suites Length: 42
      Cipher Suites (21 suites)
      Compression Methods Length: 1
      Compression Methods (1 method)
      Extensions Length: 96
    < Extension: server_name (len=16) name=example.com
      Type: server_name (0)
      Length: 16
    < Server Name Indication extension
      Server Name list length: 14
      Server Name Type: host_name (0)
      Server Name length: 11
      Server Names: example.com
    < Extension: supported_groups (len=8)
    < Extension: ec_point_formats (len=2)
    < Extension: signature_algorithms (len=26)
    < Extension: session_ticket (len=0)
    < Extension: application_layer_protocol_negotiation (len=11)
    < Extension: extended_master_secret (len=0)
    < Extension: renegotiation_info (len=1)
      [JA4: t12d2108h1_76e208dd3e22_2dae41c691ec]
      [JA4_r: t12d2108h1_000a,002f,0035,003c,003d,009c,009d,009e,009f,c009,c00a,c013,c014,c023,c024,c027,c028,c02b,c02c,c02f,c030_000a,000b,000d,0017,0023,ff01_0004,0005,0006,00401,0..]
      [JA3 Fullstring: 771,49196-49195-49200-49199-159-158-49188-49187-49192-49191-49162-49161-49172-49171-157-156-61-60-53-47-10,0-10-11-13-35-16-23-65281,29-23-24,0]
      [JA3: 74954a4c86284d0d61c4efef92b521]

```

Image - Client Hello - Client an Proxy - Transparent - Keine Authentifizierung

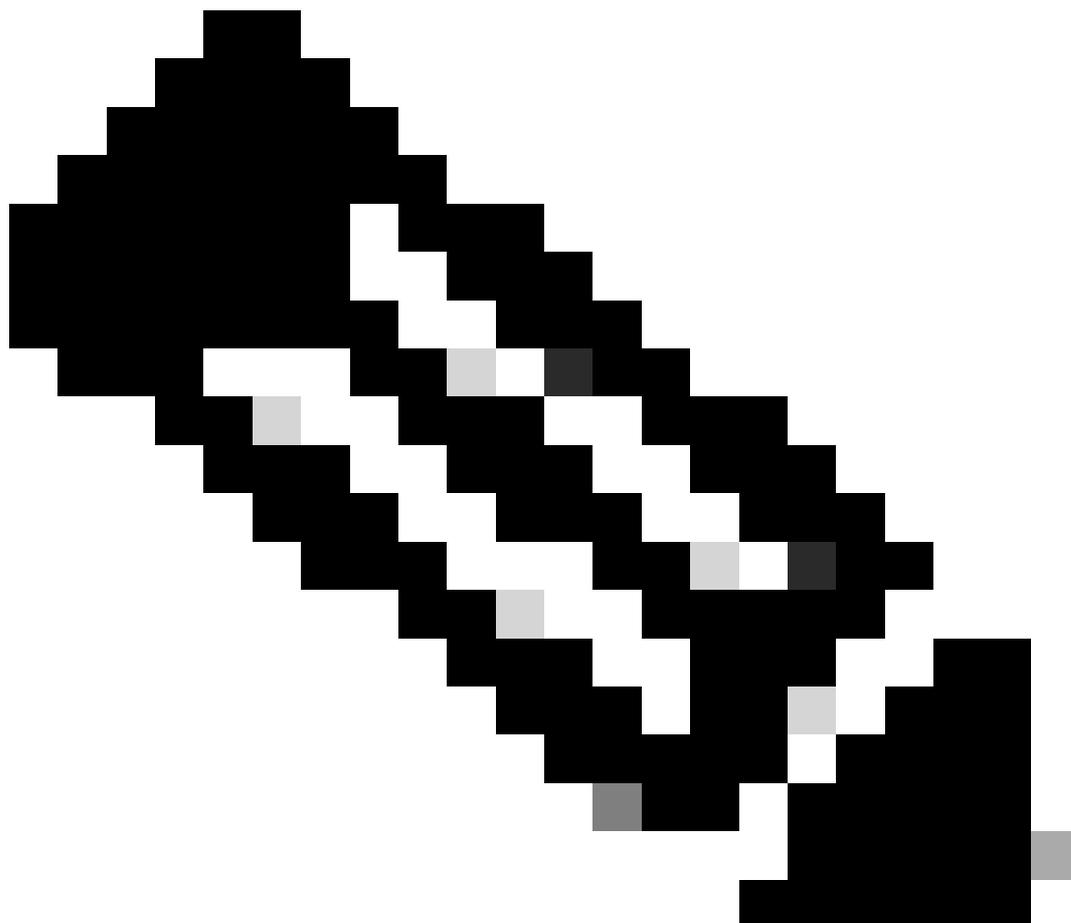


Tipp: Sie können diesen Filter in Wireshark verwenden, um nach URL/SNI zu suchen:
`tls.handshake.extensions_server_name == "www.example.com"`

Hier ist ein Beispiel für Server Key Exchange.

```
> Frame 257: 1043 bytes on wire (8344 bits), 1043 bytes captured (8344 bits)
> Ethernet II, Src: Cisco_76:fb:15 (70:70:8b:76:fb:15), Dst: Cisco_c9:c0:7f (74:88:bb:c9:c0:7f)
> Internet Protocol Version 4, Src: 93.184.216.34, Dst: 192.168.1.10
> Transmission Control Protocol, Src Port: 443, Dst Port: 54515, Seq: 1461, Ack: 189, Len: 989
> [2 Reassembled TCP Segments (2054 bytes): #256(1379), #257(675)]
Transport Layer Security
  TLSv1.2 Record Layer: Handshake Protocol: Certificate
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 2049
  Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 2045
  Certificates Length: 2042
  Certificates (2042 bytes)
    Certificate Length: 1098
  Certificate [truncated]: 308204463082032ea00302010202140440907379f2aad73d32683b716d2a7ddf2b8e2a300d06092a864886f70d01010b05003040310b30090603550406130255533110300e060355040...
  signedCertificate
    version: v3 (2)
    serialNumber: 0x0440907379f2aad73d32683b716d2a7ddf2b8e2a
    signature (sha256WithRSAEncryption)
  issuer: rdnSequence (0)
  rdnSequence: 4 items (id-at-commonName=CISCOCALO,id-at-organizationalUnitName=IT,id-at-organizationName=wsatest,id-at-countryName=US)
    > RDNSequence item: 1 item (id-at-countryName=US)
    > RDNSequence item: 1 item (id-at-organizationName=wsatest)
    > RDNSequence item: 1 item (id-at-organizationalUnitName=IT)
    > RDNSequence item: 1 item (id-at-commonName=CISCOCALO)
  validity
  subject: rdnSequence (0)
  subjectPublicKeyInfo
  extensions: 5 items
  algorithmIdentifier (sha256WithRSAEncryption)
  Padding: 0
  encrypted [truncated]: 1db2a57a8bbf4def6b1845eace5a7a17f27704e61b102f13c20a696c076bf3e736283d6cffa6c1d9417865ba7f4d4663bd3677423996e23db7f25d232eaa3110a24e72871d8cf2111d3...
  Certificate Length: 938
  Certificate [truncated]: 308203a63082028ea003020102020900a447d8363a186f2f300d06092a864886f70d01010b05003040310b30090603550406130255533110300e060355040a13077736174657374310...
Transport Layer Security
  TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange
  TLSv1.2 Record Layer: Handshake Protocol: Server Hello Done
```

Image- Serverschlüsselaustausch - Client zu Proxy - Transparent - Keine Authentifizierung



Hinweis: Wie Sie sehen, wurde das Zertifikat in SWA als Entschlüsselungszertifikat konfiguriert.

SWA und Webserver

Der Netzwerkverkehr erfolgt zwischen der IP-Adresse des Proxys und der IP-Adresse des Webservers.

Der Datenverkehr von SWA ist für TCP-Port 443 bestimmt (nicht für den Proxy-Port)

- TCP-Handshake
- TLS Handshake Client Hello - Server Hello - Serverschlüsselaustausch - Clientschlüsselaustausch
- Datenübertragung
- TCP-Verbindungsabschluss (4-Wege-Handshake)

No.	Time	Source	src MAC	Destination	dst MAC	Protocol	Length	Stream	Info
278	2023-12-11 19:36:24.251460652	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	74	17	47868 → 443 [SYN] Seq=0 Win=12288 Len=0 MSS=1460 WS=64 SACK_PERM TSval=1563255033 TSecr=0
279	2023-12-11 19:36:24.128841753	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	74	17	443 → 47868 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=64 SACK_PERM TSval=3980365294 TSecr=3980365294
280	2023-12-11 19:36:24.162744564	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=1 Ack=1 Win=13184 Len=0 TSval=1563255033 TSecr=3980365294
281	2023-12-11 19:36:24.131819080	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TLSv1	263	17	Client Hello (SNI=example.com)
282	2023-12-11 19:36:24.141189526	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=1 Ack=198 Win=65280 Len=0 TSval=3980365294 TSecr=1563255033
283	2023-12-11 19:36:24.178552585	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TLSv1	1514	17	Server Hello
284	2023-12-11 19:36:24.177104873	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=198 Ack=1449 Win=11776 Len=0 TSval=1563255183 TSecr=3980365444
285	2023-12-11 19:36:24.184184451	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	1514	17	443 → 47868 [ACK] Seq=1449 Ack=198 Win=65280 Len=1448 TSval=3980365444 TSecr=1563255033 [TCP
286	2023-12-11 19:36:24.219603043	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=198 Ack=2897 Win=10368 Len=0 TSval=1563255193 TSecr=3980365444
287	2023-12-11 19:36:24.1314885984	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TLSv1	736	17	Certificate, Server Key Exchange, Server Hello Done
288	2023-12-11 19:36:24.143459740	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=198 Ack=3567 Win=9728 Len=0 TSval=1563255193 TSecr=3980365444
289	2023-12-11 19:36:24.290848796	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	[TCP Window Update] 47868 → 443 [ACK] Seq=198 Ack=3567 Win=13184 Len=0 TSval=1563255193 TSecr=3980365444
290	2023-12-11 19:36:24.240102608	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TLSv1	192	17	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
291	2023-12-11 19:36:24.188262182	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=3567 Ack=324 Win=65152 Len=0 TSval=3980365453 TSecr=1563255193
292	2023-12-11 19:36:24.201537142	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TLSv1	117	17	Change Cipher Spec, Encrypted Handshake Message
293	2023-12-11 19:36:24.896857	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=324 Ack=3618 Win=13184 Len=0 TSval=1563255233 TSecr=3980365493
325	2023-12-11 19:36:25.383257142	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TLSv1	111	17	Application Data
326	2023-12-11 19:36:25.162026084	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=3618 Ack=369 Win=65152 Len=0 TSval=3980365883 TSecr=1563255613
327	2023-12-11 19:36:25.246545451	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TLSv1	285	17	Application Data, Application Data
328	2023-12-11 19:36:25.171978718	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=3618 Ack=588 Win=64896 Len=0 TSval=3980365883 TSecr=1563255623
329	2023-12-11 19:36:25.283437136	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TLSv1	1514	17	Application Data
330	2023-12-11 19:36:25.244187280	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=588 Ack=5066 Win=11776 Len=0 TSval=1563255673 TSecr=3980365933
331	2023-12-11 19:36:25.442898284	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TLSv1	267	17	Application Data
332	2023-12-11 19:36:25.107021532	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=588 Ack=5267 Win=11584 Len=0 TSval=1563255673 TSecr=3980365933
333	2023-12-11 19:36:25.145965305	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TLSv1	97	17	Encrypted Alert
334	2023-12-11 19:36:25.1351396684	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [FIN, ACK] Seq=619 Ack=5267 Win=12288 Len=0 TSval=1563255773 TSecr=3980365933
335	2023-12-11 19:36:25.124463214	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=5267 Ack=619 Win=64896 Len=0 TSval=3980366034 TSecr=1563255773
336	2023-12-11 19:36:25.372950	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [ACK] Seq=5267 Ack=620 Win=64896 Len=0 TSval=3980366034 TSecr=1563255773
337	2023-12-11 19:36:25.185516308	93.184.216.34	Cisco_56:5f:44	10.201.189.180	Cisco_76:fb:16	TCP	66	17	443 → 47868 [FIN, ACK] Seq=5267 Ack=620 Win=64896 Len=0 TSval=3980366034 TSecr=1563255773
338	2023-12-11 19:36:25.1425261784	10.201.189.180	Cisco_76:fb:16	93.184.216.34	Cisco_56:5f:44	TCP	66	17	47868 → 443 [ACK] Seq=620 Ack=5268 Win=12288 Len=0 TSval=1563255773 TSecr=3980366034

Image- Proxy an Webserver - HTTPS - Transparent - Keine Authentifizierung

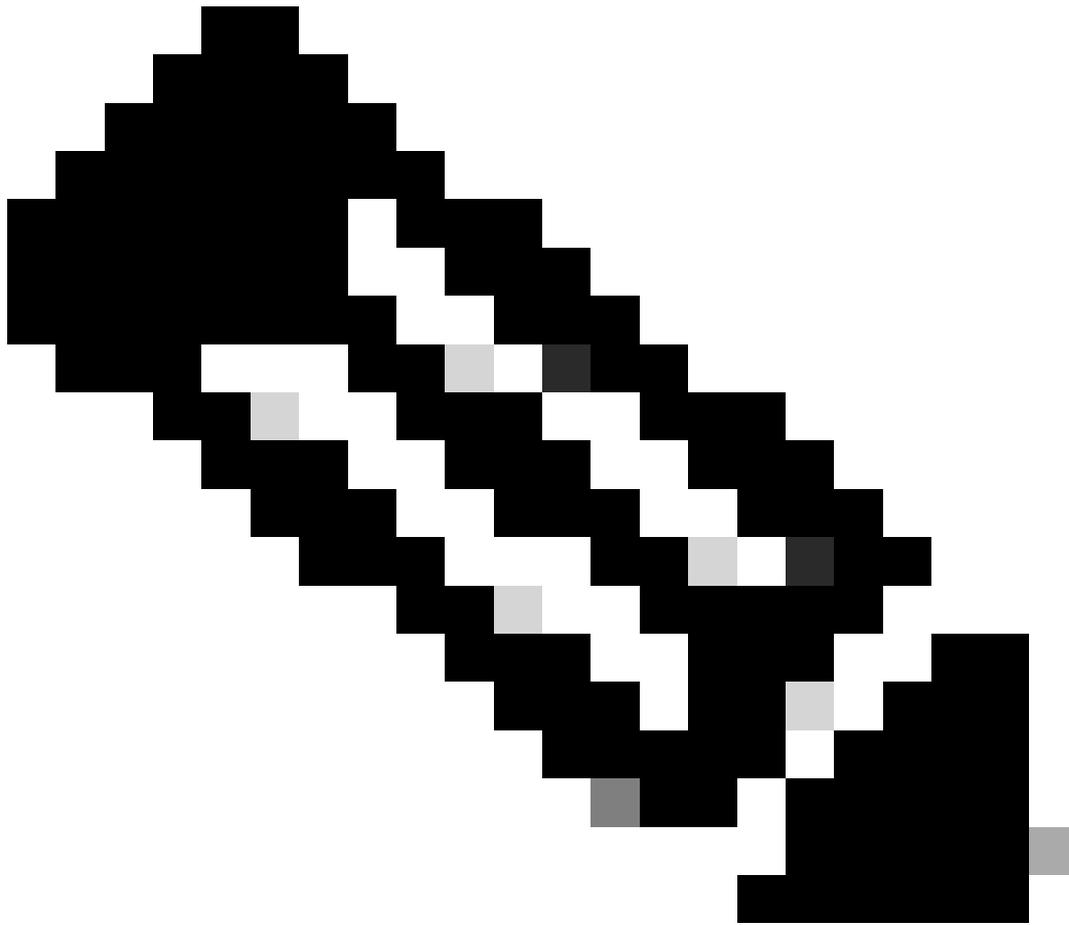
Hier ist ein Beispiel für Client Hello von SWA zu Webserver

```

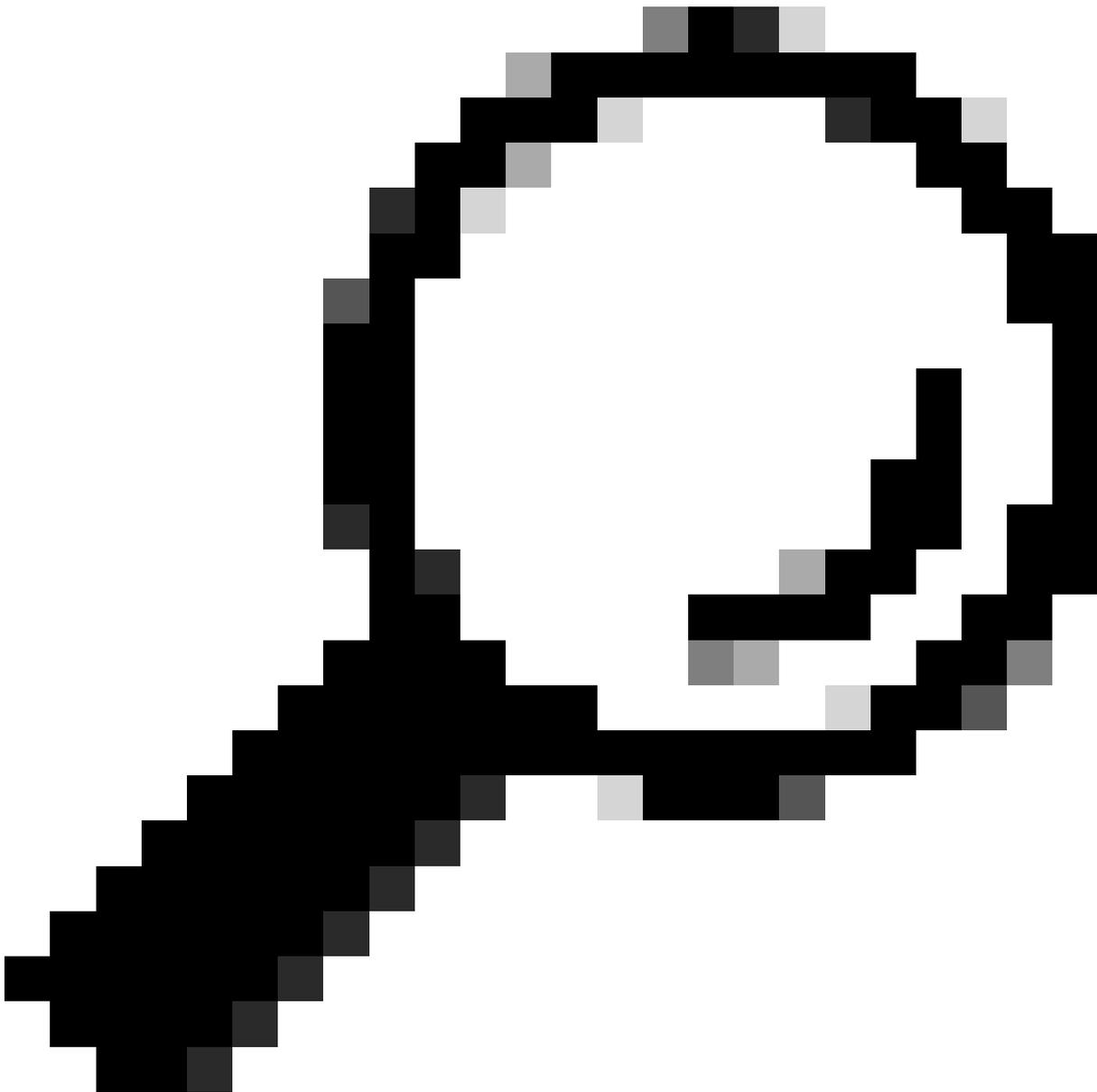
> Frame 247: 242 bytes on wire (1936 bits), 242 bytes captured (1936 bits)
> Ethernet II, Src: Cisco_c9:c0:7f (74:88:bb:c9:c0:7f), Dst: Cisco_76:fb:15 (70:70:8b:76:fb:15)
> Internet Protocol Version 4, Src: 192.168.1.10, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 54515, Dst Port: 443, Seq: 1, Ack: 1, Len: 188
> Transport Layer Security
  > TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 183
    > Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 179
      Version: TLS 1.2 (0x0303)
      > Random: 657756ab224a3f64600e99172a8d38f86b689c7eb4bb121bf54d8c96540a0f5d
      Session ID Length: 0
      Cipher Suites Length: 42
      > Cipher Suites (21 suites)
      Compression Methods Length: 1
      > Compression Methods (1 method)
      Extensions Length: 96
      > Extension: server_name (len=16) name=example.com
        Type: server_name (0)
        Length: 16
        > Server Name Indication extension
          Server Name list length: 14
          Server Name Type: host_name (0)
          Server Name length: 11
          Server Name: example.com
      > Extension: supported_groups (len=8)
      > Extension: ec_point_formats (len=2)
      > Extension: signature_algorithms (len=26)
      > Extension: session_ticket (len=0)
      > Extension: application_layer_protocol_negotiation (len=11)
      > Extension: extended_master_secret (len=0)
      > Extension: renegotiation_info (len=1)
      [JA4: t12d2108h1_76e208dd3e22_2dae41c691ec]
      [JA4_r: t12d2108h1_000a,002f,0035,003c,003d,009c,009d,009e,009f,c009,c00a,c013,c014,c023,c024,c027,c028,c02b,c02c,c02f,c030_000a,000b,000d,0017,0023,ff01_0804,0805,0806,0401,050]
      [JA3 Fullstring: 771,49196-49195-49200-49199-159-158-49188-49187-49192-49191-49162-49161-49172-49171-157-156-61-60-53-47-10,0-10-11-13-35-16-23-65281,29-23-24,0]
      [JA3: 74954a0c86284d0d6e1c4efef92b521]

```

Image - Client Hello - Proxy zu Webserver - Transparent - Keine Authentifizierung



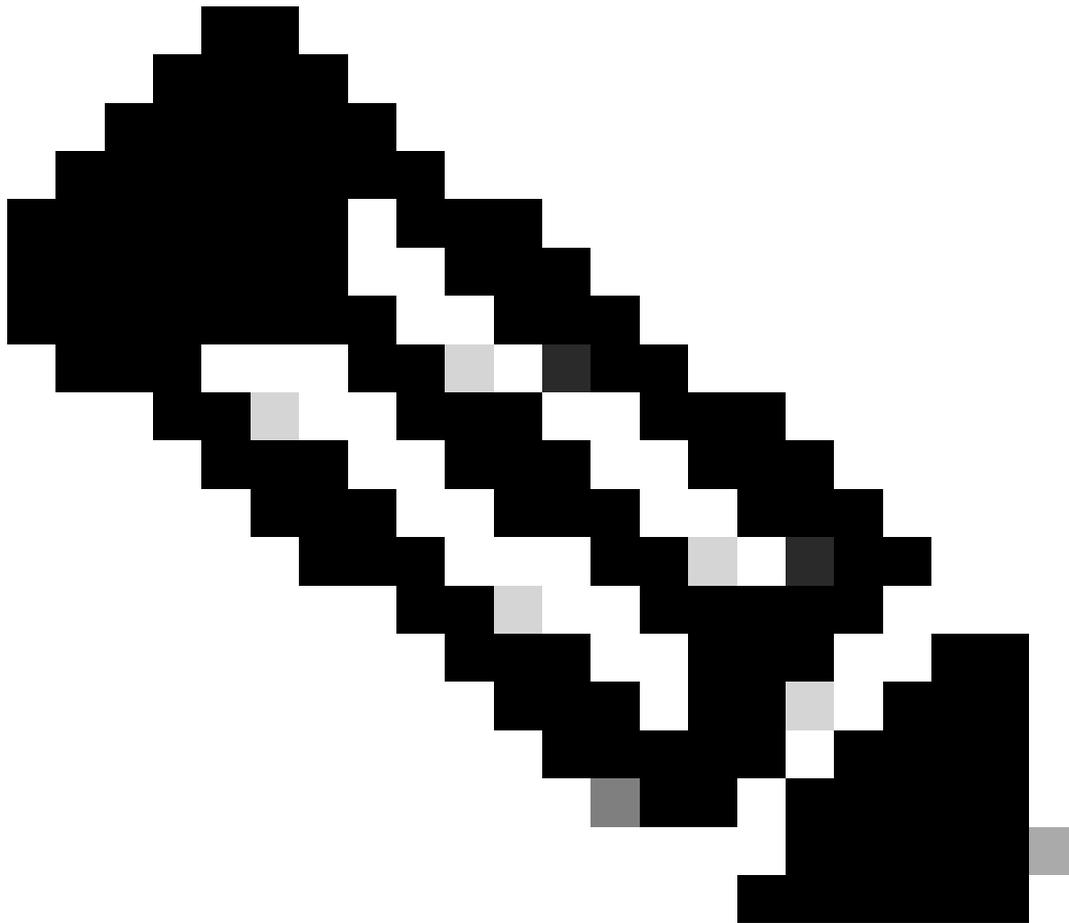
Hinweis: Die hier beobachteten Cipher Suites unterscheiden sich von den Cipher Suites in Client Hello von Client zu SWA, da die SWA, die für die Entschlüsselung dieses Datenverkehrs konfiguriert ist, ihre eigenen Ciphers verwendet.



Tipp: Im Fenster Server Key Exchange (Serverschlüsselaustausch von SWA zu Webserver) wird das Webserverzertifikat angezeigt. Wenn ein Upstreamproxy jedoch eine Konfiguration für Ihren SWA findet, wird sein Zertifikat anstelle des Webserverzertifikats angezeigt.

Beispiel für AccessLogs:

```
1702319784.943 558 192.168.1.10 TCP_MISS_SSL/200 0 TCP_CONNECT 10.184.216.34:443 - DIRECT/www.example.com
1702319785.190 247 192.168.1.10 TCP_MISS_SSL/200 1676 GET https://www.example.com:443/ - DIRECT/www.example.com
```



Hinweis: Wie Sie bei einer transparenten Bereitstellung für HTTPS-Datenverkehr sehen können, gibt es in AccessLogs zwei Zeilen. Die erste Zeile lautet, wenn der Datenverkehr verschlüsselt ist, und Sie können TCP_CONNECT und die IP-Adresse des Webserver sehen. Wenn die Entschlüsselung in SWA aktiviert ist, enthält die zweite Zeile GET, und die gesamte URL beginnt mit HTTPS. Das bedeutet, der Datenverkehr wurde entschlüsselt, und SWA kennt die URL.

Zugehörige Informationen

- [Technischer Support und Dokumentation für Cisco Systeme](#)
- [Konfigurieren von Leistungsparametern in Zugriffsprotokollen - Cisco](#)

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.