

了解Finesse的跨来源资源共享(CORS)

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[什么是CORS](#)

[CORS的生命周期](#)

[CORS与Cisco Finesse配合使用](#)

[示例：使用实时数据小工具分析CORS行为](#)

[用于CORS连接测试的TAC工具](#)

简介

本文档对跨源资源共享进行了全面介绍，以便在进行故障排除时，能够全面了解底层进程。

先决条件

要求

Cisco 建议您了解以下主题：

- Cisco Unified Contact Center Enterprise(UCCE)版本12.6.X
- 思科套装联络中心企业版(PCCE)版本12.6.X
- 思科Finesse版本12.6.X
- 思科统一情报中心(CUIC)版本12.6.X

使用的组件

本文档中的信息基于以下软件和硬件版本：

- UCCE版本12.6.2
- Finesse版本12.6.2
- CUIC版本12.6.2

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

什么是CORS

跨源资源共享(CORS)是服务器控制允许哪些网站(域、协议和端口)访问其资源的方法。虽然浏览器通常阻止来自不同来源(相同来源策略)的请求,但CORS使服务器能够选择性地放宽此限制。实质上,服务器使用特殊的HTTP报头来告诉浏览器允许哪些来源、允许哪些类型的请求(如GET、POST等),以及可以包含哪些自定义报头。这样,服务器就可以决定谁可以访问其API以及如何访问,范围从完全开放到严格限制。CORS的工作方式是让浏览器和服务器通过这些HTTP报头进行通信来管理跨源请求。

CORS使用HTTP报头启用受控跨源请求。浏览器和服务器通过这些报头进行通信,服务器指定允许的来源、方法和报头。如果服务器的响应报头缺失或无效,浏览器将阻止响应,实施相同来源策略。对于某些请求,浏览器首先向服务器发送印前检查请求,以确保它接受实际的跨源请求。

浏览器使用印前检查请求来检查服务器是否允许跨源请求,然后再发送实际请求。这些印前检查请求包括HTTP方法和自定义报头等详细信息。然后,启用CORS的服务器可以响应,允许或拒绝实际请求。如果未为CORS配置服务器,则服务器无法正确响应印前检查,并且浏览器会阻止实际请求,从而保护服务器免受不必要的跨源访问。

跨源资源共享(CORS)对Web安全和功能至关重要。它允许对不同来源(域、协议、端口)的资源进行受控访问,这是必要的,因为浏览器会实施通常阻止此类访问的相同来源策略。

CORS的生命周期

CORS请求包括两个方面:发出请求的客户端和接收请求的服务器。在客户端,开发人员编写JavaScript代码以将请求发送到服务器。服务器通过设置特定于CORS的报头来响应请求,以指示允许跨源请求。没有客户端和服务器的参与,CORS请求将失败。

CORS请求中的关键参与者是客户端、浏览器和服务器。客户端需要来自服务器的某些数据,例如JSON API响应或网页内容。浏览器充当受信任的中介,用于验证客户端是否可以从服务器访问数据。

客户端:

客户端是在网站上运行的JavaScript代码片段,它负责发起CORS请求

 注意:Finesse是一种Web应用程序。它安装在服务器上,代理只需使用其Web浏览器对其进行访问,无需在客户端安装或维护插件或其他软件。正如CORS In Action with Cisco Finesse示例所示,此架构支持实时数据报告等功能。在这种情况下,Cisco Finesse live数据小工具的JavaScript代码充当客户端,而Cisco CUIC则充当CORS生命周期内的服务器。基本上,基于浏览器的Finesse客户端与CUIC服务器交互以检索实时数据。

客户端与用户:

有时,客户端和用户这两个词可以互换使用,但在CORS环境中它们有所不同。用户是访问网站的人或在此上下文中访问Finesse的Finesse用户(座席或主管),而客户端是该网站提供的实际代码。多个用户可以访问同一网站,并可使用同一JavaScript客户端代码。

浏览器：

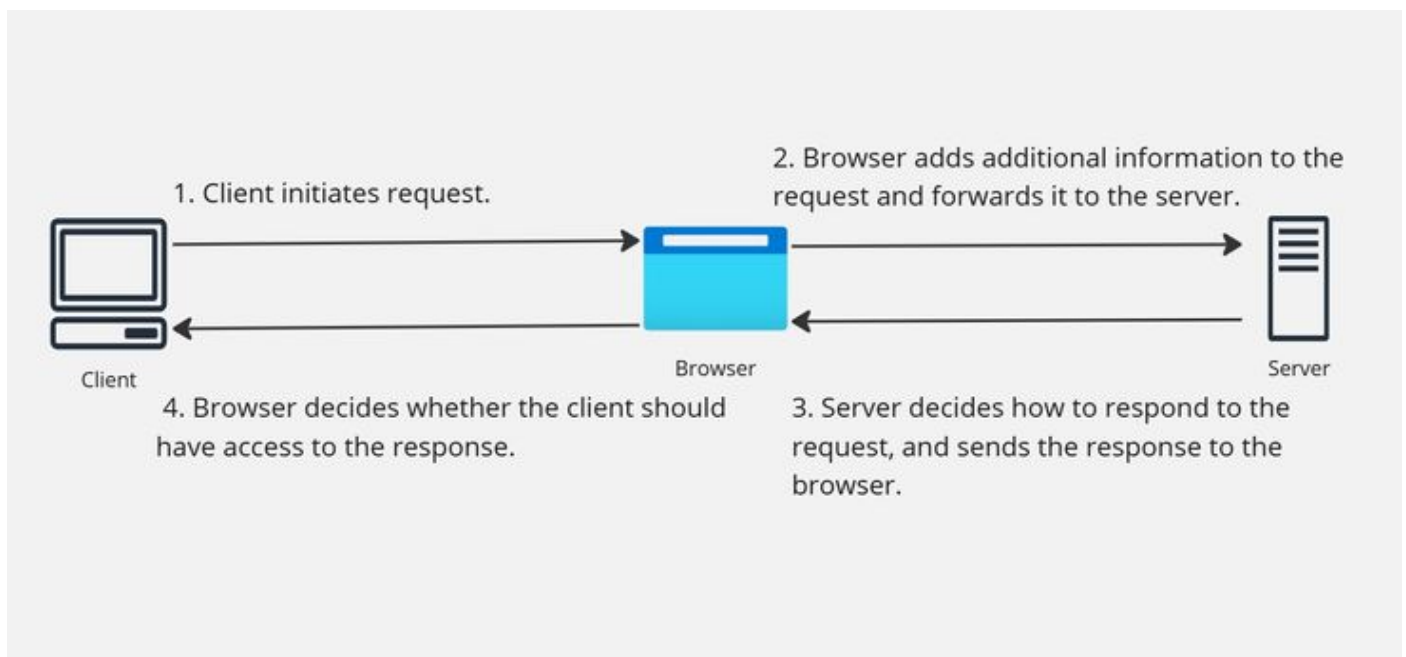
浏览器也称为用户代理，承载客户端代码。它通过向传出请求添加额外信息，使服务器能够识别客户端，从而在CORS中起到关键作用。此外，浏览器解释服务器的响应，确定是将数据传送到客户端还是返回错误。这些浏览器端操作对于维护相同来源策略提供的安全至关重要。如果浏览器不执行CORS规则，客户端可能会发出未经授权请求，从而破坏这种重要的安全机制。

服务器：

服务器是CORS请求的目标，它是Cisco Finesse的CUIC for Live data小工具示例。服务器存储客户端想要的数据，并且它对是否允许CORS请求具有最终决定权。

现在您已经知道谁参与了CORS请求，让我们看一看他们如何协同工作。后续图像说明高级CORS生命周期：

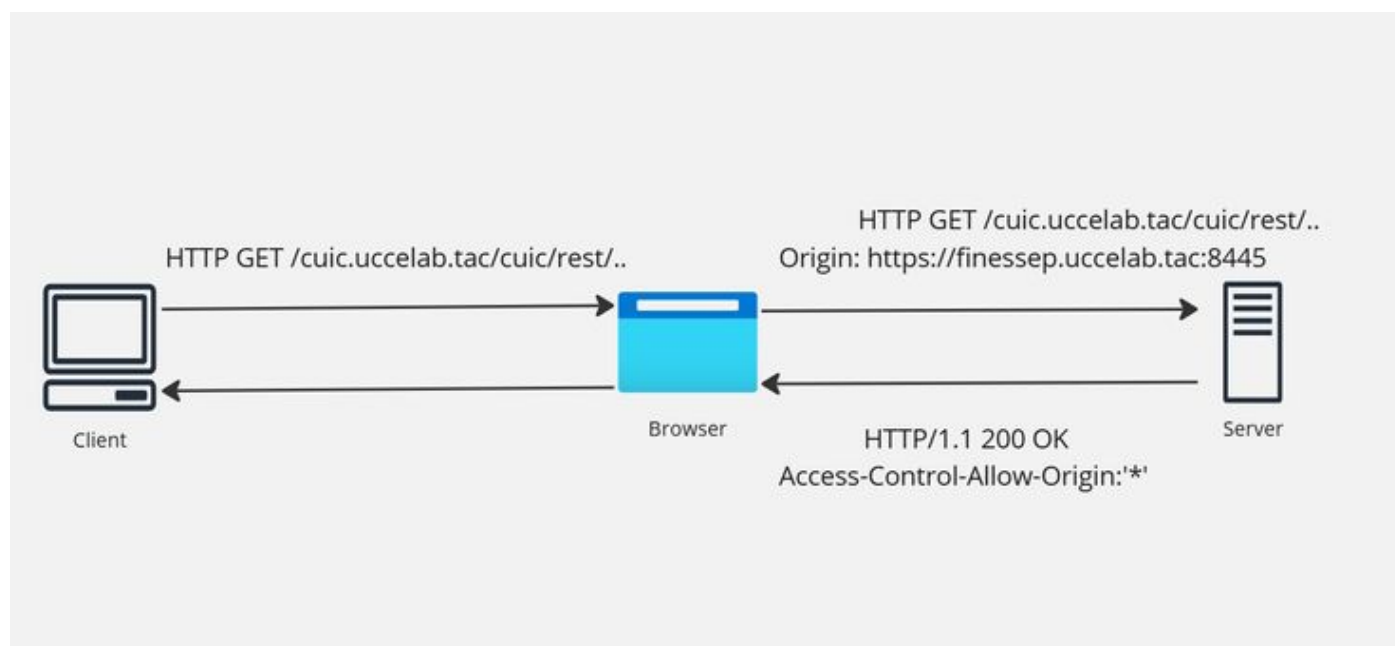
- 1.客户端发起请求。
- 2.浏览器向请求添加其他信息并将其转发到服务器。
- 3.服务器决定如何响应请求，并将响应发送到浏览器。
- 4.浏览器确定客户端是否必须有权访问响应，并将响应传递给客户端或返回错误。



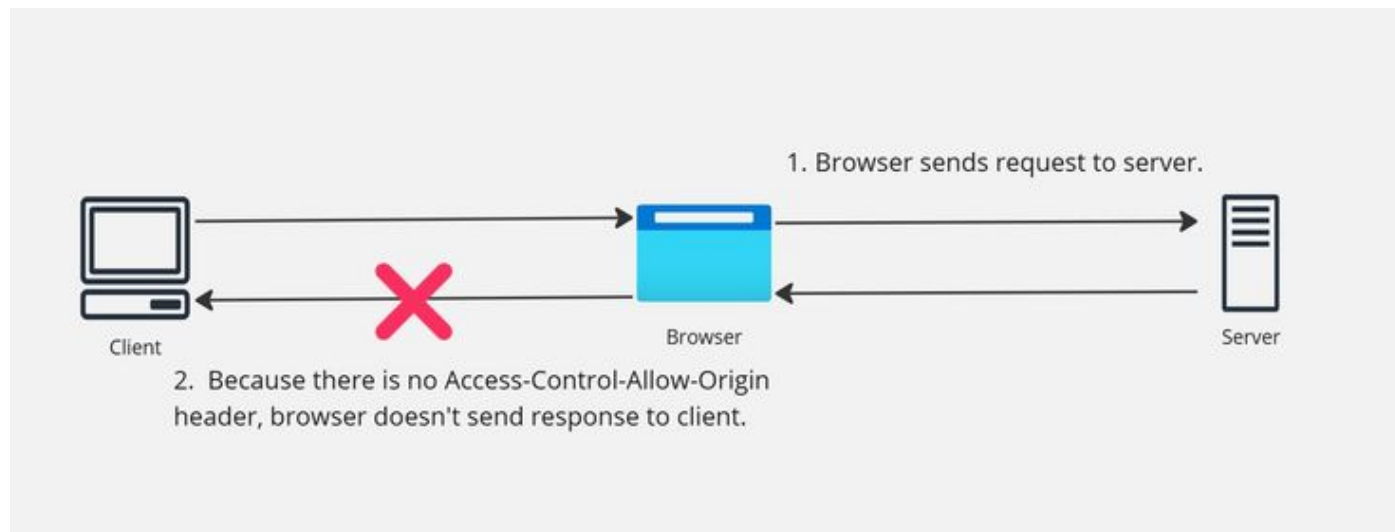
在发送跨源请求之前，浏览器会自动向HTTP请求添加源报头。客户端无法修改的报头是CORS的重要组成部分，用于标识客户端的来源（即加载客户端资源的域、协议和端口）。此安全措施可防止客户端模拟其他源。源报头是CORS的基础，因为它是客户端向服务器告知源头的方法。

在跨源资源共享(CORS)交互中，客户端的源由初始请求中的源头标识。然后，服务器在其响应中使用Access-Control-Allow-Origin报头来指示是否允许客户端访问请求的资源。此响应报头至关重要；如果没有，则CORS请求失败。Access-Control-Allow-Origin报头可以包含通配符(*)（允许来自任何来源的访问），也可以包含特定来源（仅授予该特定客户端的访问权限）。当图像显示Access-Control-Allow-Origin时：*暗示CUIC允许所有来源，CUIC通常在真实场景中发送带有特定来源的此

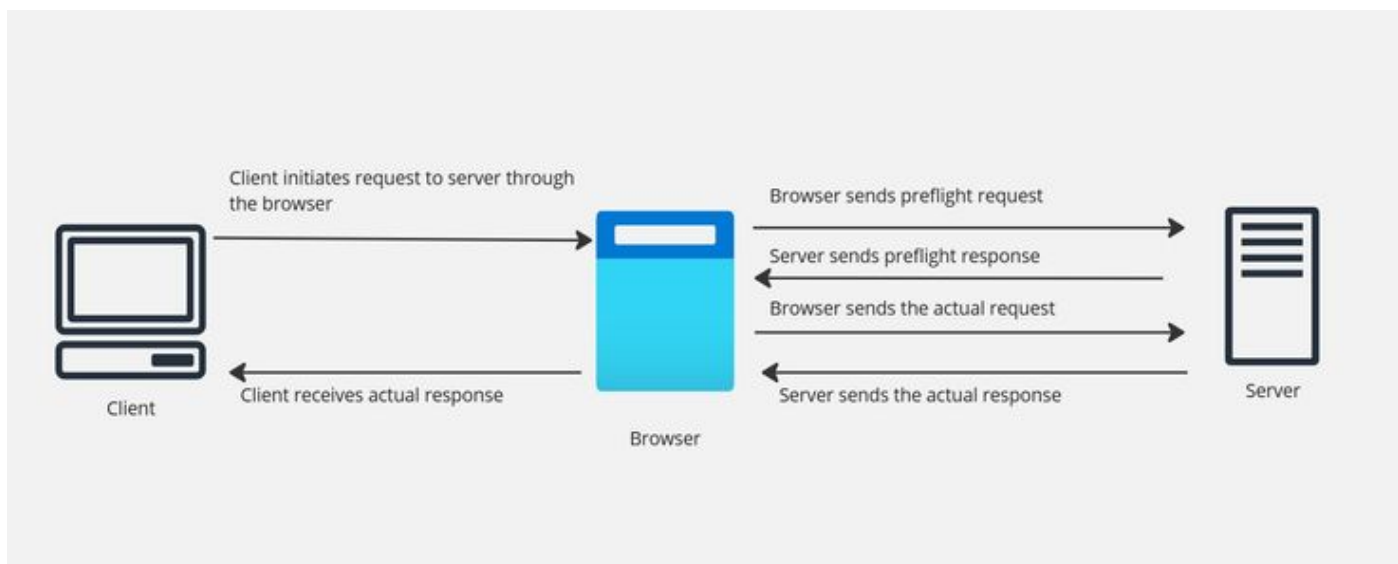
报头。



当浏览器拒绝CORS请求时，意味着客户端不会收到有关服务器响应的信息。客户端只知道发生了错误，但缺少有关特定问题的详细信息。由于很难将CORS故障与其他类型的错误区分开来，因此调试这些CORS错误可能会带来挑战。即使向服务器发送初始请求，如果服务器的响应缺少有效的Access-Control-Allow-Origin报头，浏览器也会阻止响应并在客户端触发错误，使客户端无法看到服务器的详细响应。



此图说明了整个CORS流程，尤其侧重于飞行前阶段，该阶段对于处理特定类型的跨源请求至关重要。



CORS与Cisco Finesse配合使用

示例：使用实时数据小工具分析CORS行为

本节介绍在联系中心使用Cisco Finesse进行跨来源资源共享(CORS)的典型情况。座席和主管通常使用Cisco Finesse访问实时数据报告（如示例图所示）。

当座席或主管点击报告小工具时，其操作会启动数据检索请求。此请求使用GET方法从Finesse应用程序的JavaScript代码（充当客户端）发送到CUIC/Live数据服务器。如SAML Tracer图像所示，浏览器首先向服务器发送印前检查请求，即前面所述的CORS生命周期。

The screenshot shows the Cisco Finesse web interface. The browser address bar displays the URL: `https://finessep.uccelab.tac:8445/desktop/container/?locale=en_US#/myStatistics`. The interface includes a sidebar with navigation options: Home, My Statistics (highlighted with a red box), and My History. The main content area shows the 'Agent Summary' table.

Agent	State	Logged On Time	Ready Time	Not Ready Time	% Not Ready Time	H
lab, agent1	Ready	17:27:27	00:13:24	17:14:02	98.7%	0

HTTP OPTIONS请求（印前检查请求）将发送到CUIC/实时数据服务器。此请求将源指定为Finesse服务器的完全限定域名(FQDN)，包括端口8445。这是座席用于访问Cisco Finesse应用的地址和端口。

SAML-tracer

GET https://cuicpub.ucclab.tac/security?1738431200084

GET https://cuicpub.ucclab.tac/livedata/security?1738431200084

GET https://cuicsub.ucclab.tac/livedata/security?1738431204035

GET https://cuicsub.ucclab.tac/security?1738431204035

GET https://cuicpub.ucclab.tac/security?1738431212114

GET https://cuicpub.ucclab.tac/livedata/security?1738431212114

GET https://cuicsub.ucclab.tac/security?1738431212115

GET https://cuicsub.ucclab.tac/livedata/security?1738431212115

OPTIONS https://cuicpub.ucclab.tac/livedata/api/snapshotRequest/agentConfig?userId=agent1&ids=5001

HTTP

OPTIONS https://cuicpub.ucclab.tac/livedata/api/snapshotRequest/agentConfig?userId=agent1&ids=5001 HTTP/1.1

Access-Control-Request-Method: GET

Origin: https://finessep.ucclab.tac:8445

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36

Sec-Fetch-Mode: cors

Sec-Fetch-Site: same-site

Sec-Fetch-Dest: empty

Referer: https://finessep.ucclab.tac:8445/

Accept-Encoding: gzip, deflate, br, zstd

Accept-Language: en-US,en;q=0.9

HTTP/1.1 200

server: nginx

date: Sat, 01 Feb 2025 17:33:34 GMT

content-type: application/octet-stream

access-control-allow-origin: https://finessep.ucclab.tac:8445

access-control-max-age: 600

access-control-allow-credentials: true

access-control-allow-methods: GET,POST,OPTIONS,PUT,DELETE

access-control-allow-headers: Content-Type,X-Requested-With,accept,Origin,Authorization,Access-Control-Request-Method,Access-Control-Request-Headers,Domain,locale,peripheralid,ldauthheader

access-control-expose-headers: Access-Control-Allow-Origin,Access-Control-Allow-Credentials,Access-Control-Allow-Methods,Access-Control-Allow-Headers,Access-Control-Max-Age

CUIC/Live Data服务器上的命令行界面(CLI)命令控制允许哪些源访问其实时数据资源。如果在这些设置中配置了Finesse服务器的源（其FQDN和端口），则代理可以在Finesse中查看实时数据小工具详细信息。

```
admin:utils live-data cors allowed_origin list
cors_allowed_origin
=====
1. https://finessep.ucclab.tac
2. https://finessep.ucclab.tac:8445
3. https://finesses.ucclab.tac
4. https://finesses.ucclab.tac:8445
```

```
admin:utils cuic cors allowed_origin list
cors_allowedorigins
=====
1. https://finessep.uccelab.tac
2. https://finesses.uccelab.tac
3. https://finesses.uccelab.tac:8445
4. https://finessep.uccelab.tac:8445
admin:
```

用于CORS连接测试的TAC工具

服务器端CORS配置错误有时会导致Cisco Finesse中的第三方或实时数据小工具出现问题。这篇文章提供了一个指向CORS快速检查小工具的链接，故障排除工具旨在帮助诊断影响Finesse小工具的跨源资源共享问题，包括实时数据显示和其他第三方集成。

从技术上讲，此小工具的工作方式是从Cisco Finesse客户端向指定的目标资源发送印前检查请求。此快速检查功能有助于快速确定和解决与CORS相关的问题，从而加快故障排除过程。

要在Finesse桌面中部署Contact Center CORS Quick Check 12.6-v1.0小工具，请执行以下操作：

- 1.从Contact Center [CORS Quick Check 12.6-v1.0](#) folder.2下载小工具的文件。
- 2.将Contact Center CORS Quick Check 12.6-v1.0文件夹的内容复制到Finesse安装中的3rdpartygadget目录中。
- 3.在Finesse桌面布局中将小工具添加到所需的用户角色（座席、主管等）。提供的示例XML演示了添加此小工具的正确配置。

```
<gadget>/3rdpartygadget/files/TestCORSgadget.xml</gadget>
```

有关上传第三方小工具并将其添加到桌面的详细信息，请参阅[Finesse开发人员指南](#)中的第三方小工具一章和[Finesse管理指南](#)中的管理第三方小工具一章。

上传小工具文件并重新启动Cisco Finesse Tomcat服务后，小工具即可使用并显示图形用户界面(GUI)。

← → ↻ Not secure https://finessep.uccelab.tac:8445/desktop/container/?locale=en_US#/GadgetTest

 Cisco Finesse  Ready 00:47:55

Contact Center CORS Quick Check

Insert FQDN/IP address of the targeted resource:

Choose which Cisco product you are testing:

Cisco CUIC

--Please choose an option--

Cisco Finesse

Cisco CUIC

Other

Home
My Statistics
My History

您可以从顶部的下拉列表中选择CUIC。在提供的字段中输入CUIC服务器的完全限定域名(FQDN)。一个成功的测试将如下所示。

← → ↻ Not secure https://finessep.uccelab.tac:8445/desktop/container/?locale=en_US#/GadgetTest

 Cisco Finesse  Ready 00:51:44

Contact Center CORS Quick Check

Insert FQDN/IP address of the targeted resource:

Choose which Cisco product you are testing:

Cisco CUIC

CORS Preflight Test Succeeded ✓

Home
My Statistics
My History

成功的测试意味着CUIC服务器已正确配置为与Finesse服务器进行跨源资源共享(CORS)。浏览器的SAML Tracer日志显示HTTP选项请求 (CORS预检) 已发送到CUIC服务器。此请求在Origin报头中包含Finesse服务器的地址。CUIC服务器以200 OK HTTP消息进行响应, 重要的是, 响应中的Access-Control-Allow-Origin报头也包含Finesse服务器的地址。这确认已将CUIC服务器配置为允许来自Finesse服务器源的请求, 从而验证CORS是否已正确设置。

<#root>

OPTIONS https://cuicpub.uccelab.tac/cuic/ HTTP/1.1

sec-ch-ua-platform: "Windows"

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome..

sec-ch-ua: "Google Chrome";v="131", "Chromium";v="131", "Not_A Brand";v="24"

sec-ch-ua-mobile: ?0

Accept: */*

Origin: https://finessep.uccelab.tac:8445

Sec-Fetch-Site: same-site

Sec-Fetch-Mode: cors

Sec-Fetch-Dest: empty

Referer: https://finessep.uccelab.tac:8445/

Accept-Encoding: gzip, deflate, br, zstd

Accept-Language: en-US,en;q=0.9

<#root>

HTTP/1.1 200

server: nginx

date: Sat, 08 Feb 2025 01:27:47 GMT

content-length: 0

strict-transport-security: max-age=31536000; includeSubDomains

set-cookie: JSESSIONID=bE73993C4A7C1Fc1b33A7AaF897B8428; Path=/cuic; Secure; HttpOnly; SameSite=Strict

pragma: No-cache

cache-control: no-cache

expires: Thu, 01 Jan 1970 00:00:00 GMT

x-frame-options: SAMEORIGIN

x-xss-protection: 1; mode=block

x-content-type-options: nosniff

content-security-policy: default-src 'self' ; script-src 'self' data: 'unsafe-inline' 'unsafe-eval' ; s

vary: origin,access-control-request-method,Access-Control-Request-Headers

access-control-allow-origin: https://finessep.uccelab.tac:8445

access-control-allow-credentials: true

access-control-expose-headers: access-control-allow-origin,access-control-allow-credentials,access-cont

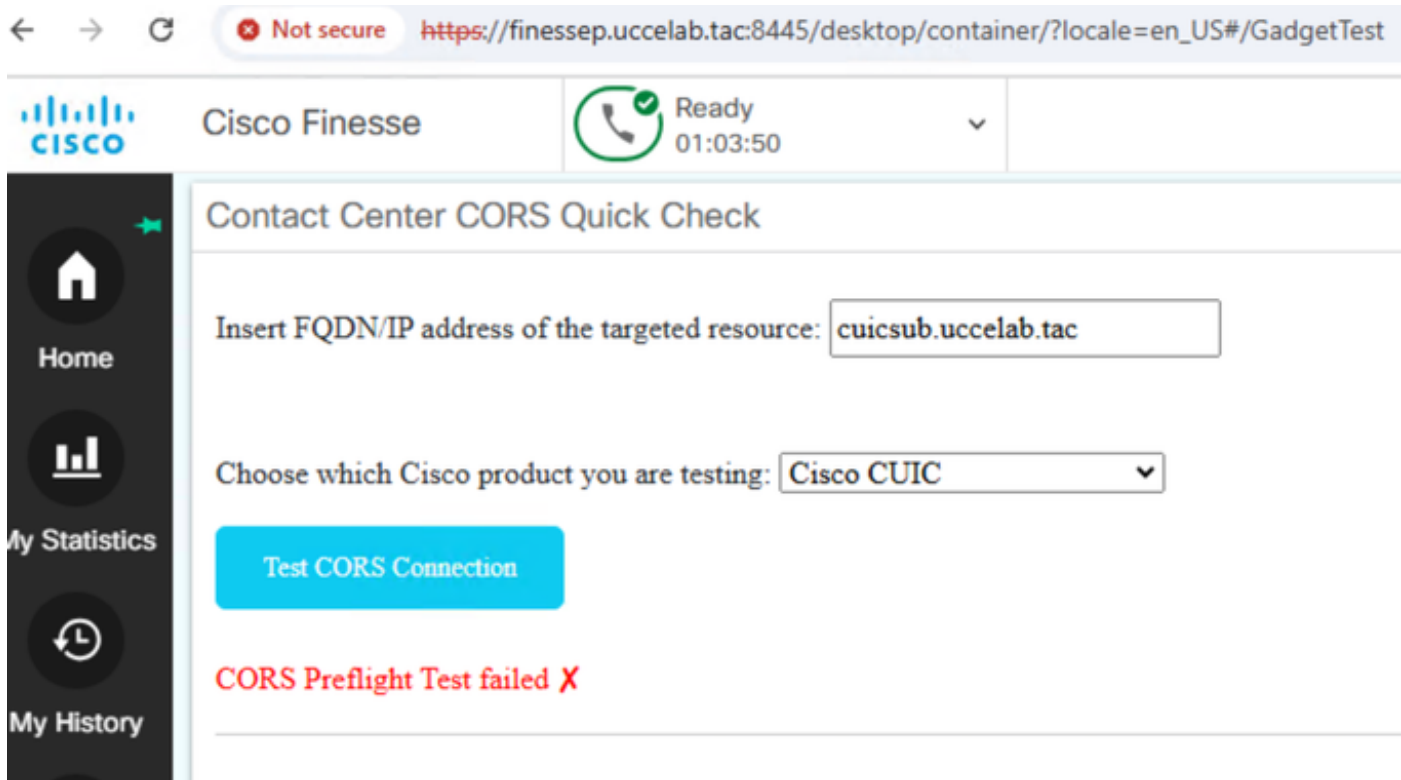
access-control-max-age: 600

access-control-allow-methods: DELETE,POST,GET,OPTIONS,PUT

access-control-allow-headers: referer,peripheralid,origin,access-control-request-method,locale,accept,a

allow: GET,POST,OPTIONS,PUT,DELETE

在此场景中，该工具演示了非工作配置。与上一个示例不同，Finesse服务器未配置为CUIC服务器上的用户。相反，它仅在CUIC发布服务器上配置。因此，CORS预检请求失败，CUIC服务器以HTTP 403(Forbidden)错误做出响应。



<#root>

OPTIONS https://cuicsub.uccelab.tac/cuic/ HTTP/1.1

Accept: */*

Access-Control-Request-Method: OPTIONS

Origin: https://finessep.uccelab.tac:8445

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome..

Sec-Fetch-Mode: cors

Sec-Fetch-Site: same-site

Sec-Fetch-Dest: empty

Referer: https://finessep.uccelab.tac:8445/

Accept-Encoding: gzip, deflate, br, zstd

Accept-Language: en-US,en;q=0.9

<#root>

HTTP/1.1 403

server: nginx

date: Sat, 08 Feb 2025 01:54:52 GMT

content-type: text/html; charset=utf-8

content-length: 2143

strict-transport-security: max-age=31536000; includeSubDomains

set-cookie: JSESSIONID=1C7606841B83d7847486c3d18D31cEfD; Path=/cuic; Secure; HttpOnly; SameSite=Strict

pragma: No-cache

cache-control: no-cache

expires: Thu, 01 Jan 1970 00:00:00 GMT

x-frame-options: SAMEORIGIN

x-xss-protection: 1; mode=block

x-content-type-options: nosniff

从CUIC用户命令行界面(CLI)的输出中可以看到，Cisco Finesse未列出。这表示Finesse当前未配置为此CUIC服务器上的用户。

<#root>

```
admin:utils cuic cors allowed_origin list
```

```
cors_allowedorigins
```

```
=====
```

1. https://finessep.ucclab.tac
2. https://finesses.ucclab.tac
3. https://finesses.ucclab.tac:8445

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。