

Understand Call Routing Logic on Meeting Server

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[What is the Call Routing Logic of the Cisco Meeting Server \(CMS\)?](#)

[Step 1. Incoming Call Matching Table](#)

[Step 2. Incoming Call Forwarding Table](#)

[Rewrite Domain](#)

[Caller ID](#)

[Step 3. Outbound Call Table](#)

[Verify](#)

[Troubleshoot](#)

[Related Information](#)

Introduction

This document describes the call routing logic of the Cisco Meeting Server (CMS) (formerly Acano product) which is split up in several call routing tables. This document covers the different stages and scenarios that calls can take through these call routing tables.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco Meeting Server Call Bridge component.


Components Used

The information in this document is based on Cisco Meeting Server on version 2.3.x.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

What is the Call Routing Logic of the Cisco Meeting Server (CMS)?


The call routing on CMS involves a few call routing different tables. With the flow chart that can be downloaded , you can follow the call routing logic for each call that arrives on the CMS. This is valid for all types of calls: Cisco Meeting App (CMA - thick client or WebRTC), Standard Session Initiation Protocol (SIP) calls or Microsoft SIP calls unless specified otherwise.


 **Note:** The only exception would be for CMS initiated calls (either CMS directly for TelePresence Management Suite (TMS) scheduled outbound calls or CMA client calls out) in which the call forwarding table is bypassed.

This is the order of call route process within CMS:

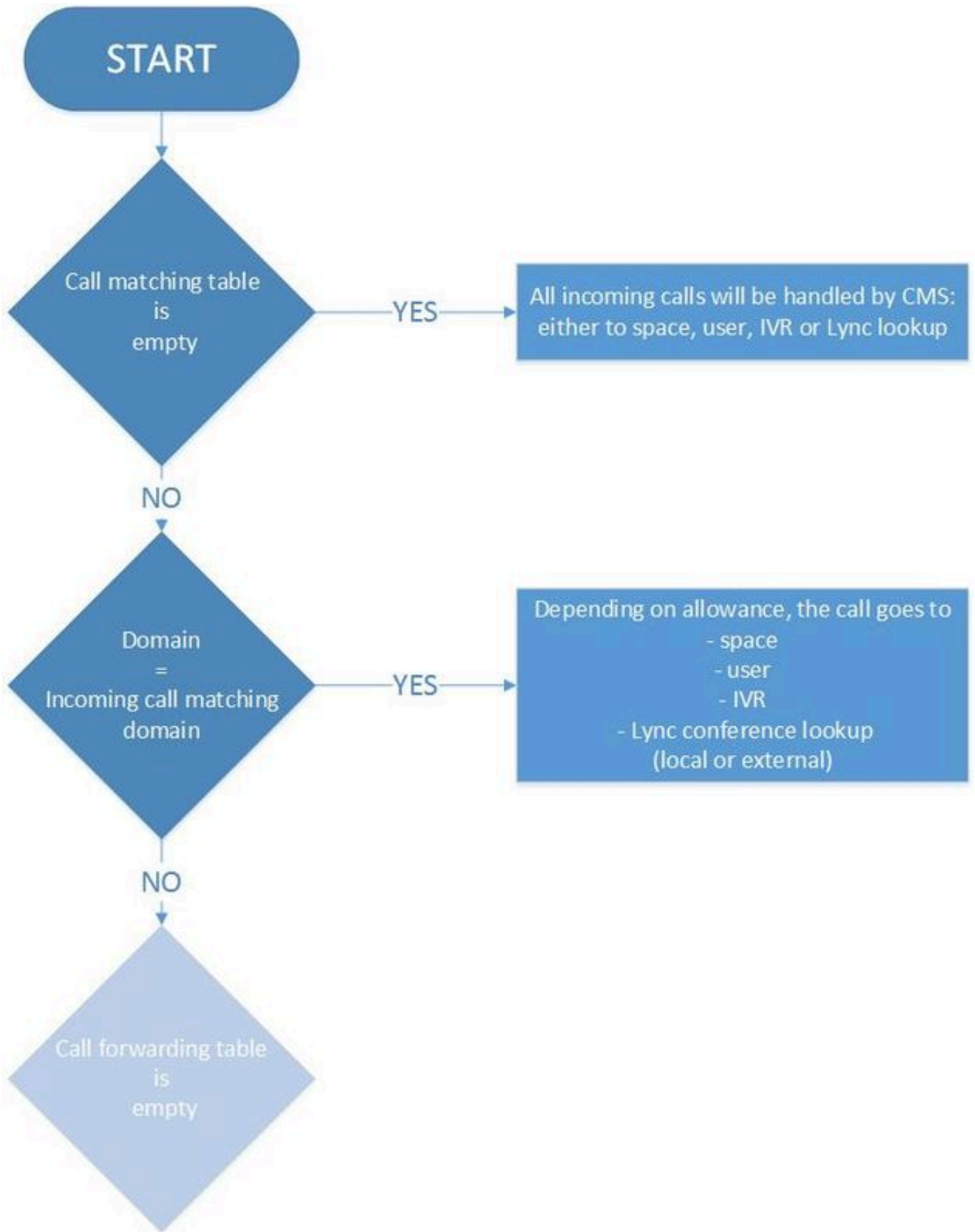
1. Incoming Call Matching Table
2. Incoming Call Forwarding Table
3. Outbound Call Table

Each table is explained in more detail later in the document, which includes the images that show only the relevant part of the .

 **Note:** CMS only performs call routing based on domain routing, thus based on the right-hand side (RHS) of the Uniform Resource Identifier (URI). There is no call routing functionality based on the left-hand side (LHS) of the URI like you have on Cisco Unified Communications Manager (CUCM) with DirectoryNumber routing (Route Patterns).


 **Note:** Each table is an ordered list set by the priority attribute. Higher priority means it tries to be matched first. If it does not match, it proceeds with the next rule in the list. As a general rule of thumb, give more general rules (like a * that matches any domain) a lower priority than the more specific rules. That way, the specific rules are handled first, and you have the fallback potentially to the more general rules.

Step 1. Incoming Call Matching Table



This is the first step in the process in which CMS determines whether the inbound call is destined for the Cisco Meeting Server itself and would need to process on it further or whether it is a call destined for a different system in which CMS is the agent that interworks the call and handles both the media and signaling (f.e. Skype gateway calls to Standard SIP endpoints or vice versa).

It checks if the domain part of the incoming URI matches the incoming matching table or not. If it does match, it is able to route the call to space, user, IVR or do a Lync conference lookup (on-prem or off-prem) as per your configuration for this dialplan rule. The table does not allow for wildcard domains, it requires a full match.

 **Note:** If you do not have any incoming call matching domains configured, CMS accepts all incoming URIs from SIP or Lync calls that land on the callbridge. For CMA clients (WebRTC or thick client) though it accepts the call, yet it is not routed to the correct space or user automatically. Thus, it is important to enter in the correct domain when you use the CMA client to dial to spaces or users in this case.

For example, a call matching table is shown in the image (it only shows the **Targets spaces** and **Targets users** option for brevity):

Incoming call handling

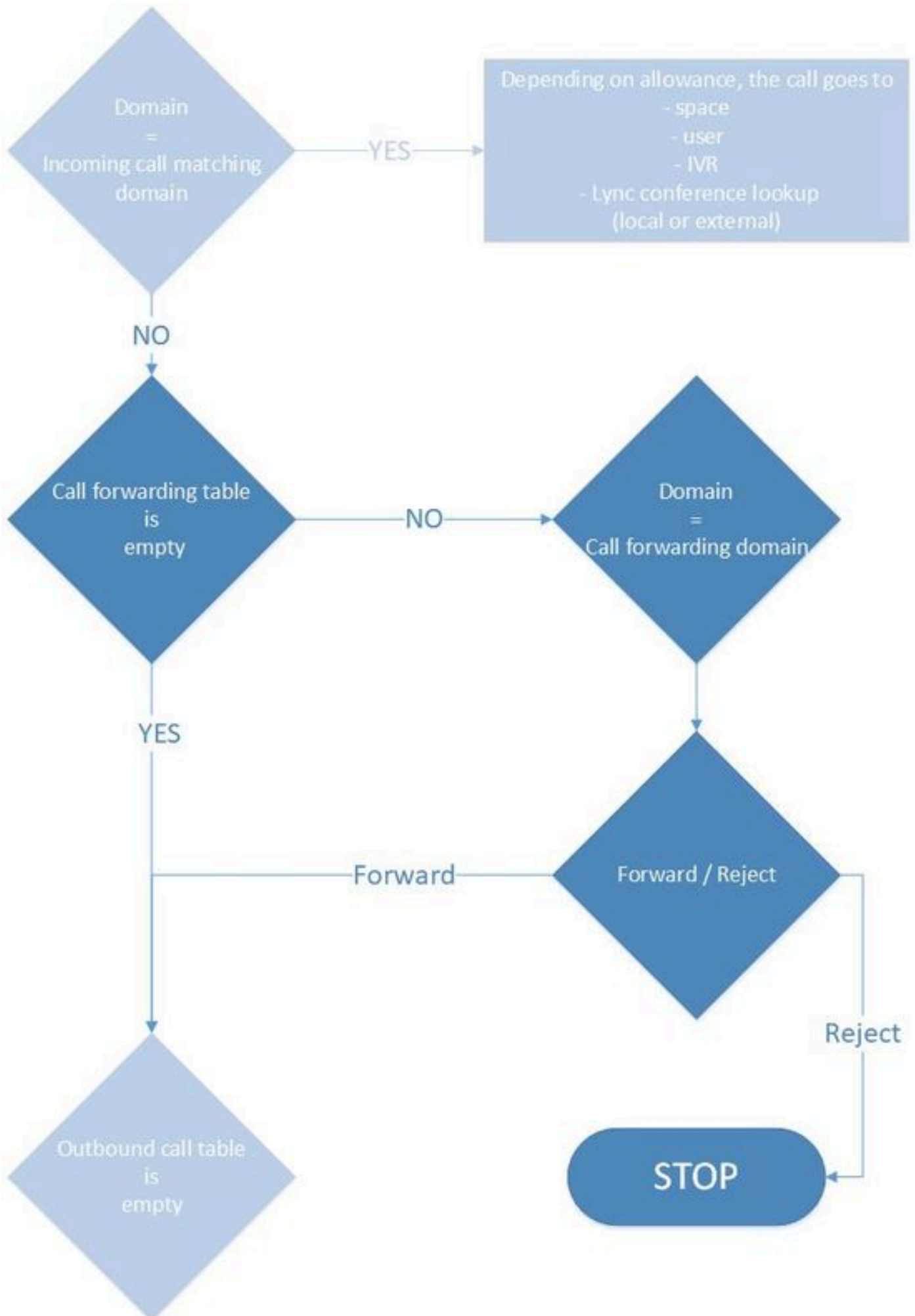
Call matching

| <input type="checkbox"/> | Domain name | Priority | Targets spaces | Targets users |
|--------------------------|-------------------------|----------------------|----------------|---------------|
| <input type="checkbox"/> | acano.steven.lab | 2 | yes | yes |
| <input type="checkbox"/> | 10.48.54.160 | 1 | yes | yes |
| <input type="checkbox"/> | acano1.acano.steven.lab | 0 | yes | yes |
| <input type="checkbox"/> | <input type="text"/> | <input type="text"/> | yes ▾ | yes ▾ |


1

Here the domain is set up as acano.steven.lab which the clients normally dial. However, it also allows for ad-hoc calls or specific SIP route patterns from CUCM (or Expressway search rules) that only target a specific callbridge (in case of a cluster) by the first and second fallback rule in the table that match either the IP address of the callbridge (10.48.54.160 in this case) or the Fully Qualified Domain Name (FQDN) of the callbridge (acano1.acano.steven.lab in this case).

Step 2. Incoming Call Forwarding Table



If the call did not hit any of the rules on the incoming call matching table or there was no match for the call t

 : This does happen though with CMA clients (thick clients and WebRTC) as they are able to make outbound calls (*Web App in 3.0 cannot make outbound calls, but rather calls made by CMS space out by Callbridge). Similarly, outbound calls on CMS do work fine as well when made via API for example (in the case of TMS scheduled conferences). In general, calls that are initiated from CMS itself (either CMS directly or via CMA) must not follow the call forwarding logic.

In the event log, you can see the highlighted **forwarding** message as for example when CMS acts as a gateway for SIP and Skype calls. Just before that, you can see the **incoming** call and the **outgoing** call afterwards.

<#root>

2018-10-04 06:36:24.612 Info call 788:

incoming

SIP call from "sip:1060@10.48.36.215" to local URI "sip:stejanss@any.com"

2018-10-04 06:36:24.624 Info

forwarding call

to 'sip:stejanss@any.com' to 'stejanss@any.com'

2018-10-04 06:36:24.625 Info call 789:

outgoing

SIP call to "stejanss@any.com"

If the forwarding table does not have any rule or a reject rule, then the event log does not explicitly show this. It just informs you that the SIP call did not match (any space, user, IVR or Lync meeting) and that you miss the forwarding rule (or it is set to reject) to move to the outbound rules section.

<#root>

2018-10-04 06:47:12.482 Info call 790:

incoming

SIP call from "sip:1060@10.48.36.215" to local URI "sip:stejanss@any.com"

2018-10-04 06:47:12.495 Info call 790: ending; local teardown, destination URI not matched - not

For CMA clients calls or outbound calls from CMS that are initiated through TMS scheduled meetings, there is **no incoming** call seen in the event log. The call immediately goes to the outbound dial plan table and is not processed by the call forwarding table.

In the call forwarding table, there are two other configuration options: Rewrite Domain and Caller ID.

Rewrite Domain

This option allows you to rewrite the domain of the inbound call to a different one and changes the domain part of the **SIP Request-URI** as well as the **To** header of the SIP message.

| Domain matching pattern | Priority | Forward | Caller ID | Rewrite domain | Forwarding domain |
|--------------------------------------|----------|---------|---------------|----------------|-------------------|
| <input type="checkbox"/> any.com | 2 | forward | use dial plan | yes | newany.com |
| <input type="checkbox"/> dummy.com | 0 | reject | use dial plan | no | |
| <input type="checkbox"/> tpiab.local | 0 | forward | use dial plan | no | |
| <input type="text"/> | 0 | reject | use dial plan | no | |

1
Delete

For example, with the configuration on this image, the event log (with SIP trace enabled) is shown here for an inbound call with domain **any.com** but without a match on the incoming call matching table (either on spaces, users, IVR or Skype conferences):

<#root>

2018-10-04 07:02:24.818 Info SIP trace: connection 0: incoming SIP TCP data from 10.48.36.215:564
 2018-10-04 07:02:24.818 Info SIP trace:

INVITE

sip:stejanss@

any.com

SIP/2.0

2018-10-04 07:02:24.818 Info SIP trace: Via: SIP/2.0/TCP 10.48.36.215:5060;branch=z9hG4bK53e4c4ce
 2018-10-04 07:02:24.818 Info SIP trace: From: "EX60 Steven" <sip:1060@steven.lab>;tag=742103~ee54
 2018-10-04 07:02:24.818 Info SIP trace:

To:

<sip:stejanss@

any.com

>

..
 2018-10-04 07:02:24.822 Info call 797:

incoming

SIP call from "sip:1060@10.48.36.215" to local URI "sip:stejanss@

any.com

"

2018-10-04 07:02:24.834 Info

forwarding

call to 'sip:stejanss@

any.com

' to 'stejanss@

newany.com

,

2018-10-04 07:02:24.835 Info call 798:

outgoing

SIP call to "stejanss@

newany.com

"

```

..
2018-10-04 07:02:24.838 Info SIP trace: connection 19: outgoing SIP TCP data to 10.48.36.215:5060
2018-10-04 07:02:24.838 Info SIP trace:

INVITE

sip:stejanss@

newany.com

SIP/2.0
2018-10-04 07:02:24.838 Info SIP trace: Via: SIP/2.0/TCP 10.48.80.71:5060;branch=z9hG4bKefc98b81a
2018-10-04 07:02:24.839 Info SIP trace: Call-ID: 18644f28-e998-4032-a7df-75325e9d11b0
2018-10-04 07:02:24.839 Info SIP trace: CSeq: 659590315 INVITE
2018-10-04 07:02:24.839 Info SIP trace: Max-Forwards: 70
2018-10-04 07:02:24.839 Info SIP trace: Contact: <sip:1060@10.48.80.71;transport=tcp>
2018-10-04 07:02:24.839 Info SIP trace:

To

: <sip:stejanss@

newany.com

>
2018-10-04 07:02:24.839 Info SIP trace: From: "EX60 Steven" <sip:1060@steven.lab>;tag=2aa2a49bba2

```

In this forwarding call line, it shows the modification that has happened. In case you do not have SIP trace enabled, it still shows the modification of any.com to newany.com.

The most common use of this rewrite of the domain comes with an on-prem [Lync integration with a CMS cluster](#) where it is recommended to set the Contact header and From header in the outbound rules to Lync/Skype to the callbridge specific Fully Qualified Domain Names (FQDNs). That is because of these routing rules:

- Skype sends new **transactions** within a dialog (like for example an ACK after an INVITE - 200 OK) to the **Contact** header specified in the 200 OK it received from the CMS. For inbound connections from Skype to CMS, Skype first sends a NEGOTIATE SIP message containing a ms-fe header in the To header which specifies how the Contact header must be filled in in the 200 OK replies on the INVITE (as it uses the same TCP channel)
- Skype sends new **dialogs** (like content sharing, as it is a separate call, or a callback in case of a missed call) to the **From** header of the original INVITE

As it rewrites the domain, it is relevant for the callback from Lync calls. The From header of the missed INVITE, points to the specific callbridge where the call comes from. Lync then sends a new request (INVITE) with SIP Request URI which matches the callbridge FQDN. It is then translated to the SIP domain through these rewrite rules. Once the call is forwarded it uses the outbound rules towards the CUCM or Expressway-C where the SIP endpoint is registered.

Caller ID

There are two options here that can be set on the forwarding rules. Either it is set to **pass through** and then no modification is made on the From header of the outbound INVITES or it is set to **use dial plan** which allows the system to modify the From header as per the outbound rules. This setting is irrespective of the fact whether you have a rewrite of the domain as that just concerns the SIP Request URI as well as the To header of the outbound INVITE.

As an example, the same call as before was made but now there is an outbound dial plan rule to newany.com (as after the rewrite on the incoming call forwarding table) set up as a Lync type call (Ms-Conversation-ID as extra SIP header for example). Appropriately, the Local From Domain (and Local Contact Domain) are filled in to point to the callbridge FQDN as indicated earlier for Lync calls. This then reflects the change on **From** and **Contact** header of the outbound SIP INVITE. As shown in the image, they are populated with the same value and can be selected individually as per your requirement.

Outbound calls

| Filter | Domain | SIP proxy to use | Local contact domain | Local from domain | Trunk type | Behavior | Priority |
|--------------------------|------------|------------------|------------------------|----------------------------|--------------|----------|----------|
| <input type="checkbox"/> | stevan.lab | 10.48.36.46 | | <use local contact domain> | Standard SIP | Stop | 5 |
| <input type="checkbox"/> | newany.com | 10.48.36.46 | callbridgefqdn.any.com | callbridgefqdn.any.com | Lync | Stop | 4 |

<#root>

```
2018-10-12 09:09:24.488 Info SIP trace: connection 28: incoming SIP TCP data from 10.48.36.215:44
2018-10-12 09:09:24.489 Info SIP trace: INVITE sip:stejanss@any.com SIP/2.0
2018-10-12 09:09:24.489 Info SIP trace: Via: SIP/2.0/TCP 10.48.36.215:5060;branch=z9hG4bKf4a230ec
2018-10-12 09:09:24.489 Info SIP trace:
```

From

: "EX60 Steven" <sip:1060@

stevan.lab

>;tag=118288~ee545a46-516a-4de6-87d7-7b1f5a5b848a-32900729

```
2018-10-12 09:09:24.489 Info SIP trace: To: <sip:stejanss@any.com>
2018-10-12 09:09:24.489 Info SIP trace: Call-ID: 81e67f80-bc0164c4-f2c6-d724300a@10.48.36.215
```

```
2018-10-12 09:09:24.494 Info call 803:
```

incoming

```
SIP call from "sip:1060@10.48.36.215" to local URI "sip:stejanss@any.com"
2018-10-12 09:09:24.506 Info
```

forwarding call

```
to 'sip:stejanss@any.com' to 'stejanss@newany.com'
2018-10-12 09:09:24.507 Info call 804:
```

outgoing

SIP call to "stejanss@newany.com" (Lync)

```
2018-10-12 09:09:24.507 Info SIP trace: connection 33: allocated for outgoing connection to 10.48
2018-10-12 09:09:24.508 Info SIP trace: connection 33: outgoing connection successful, 10.48.80.7
2018-10-12 09:09:24.510 Info SIP trace: connection 33: outgoing SIP TCP data to 10.48.36.46:5060
2018-10-12 09:09:24.510 Info SIP trace: INVITE sip:stejanss@newany.com SIP/2.0
2018-10-12 09:09:24.510 Info SIP trace: Via: SIP/2.0/TCP 10.48.80.71:5060;branch=z9hG4bK15bdde97a
2018-10-12 09:09:24.510 Info SIP trace: Call-ID: c366ddaf-e602-4fa5-b1d6-2e16ec08534a
2018-10-12 09:09:24.510 Info SIP trace: CSeq: 1498747095 INVITE
2018-10-12 09:09:24.510 Info SIP trace: Max-Forwards: 70
2018-10-12 09:09:24.510 Info SIP trace:
```

Contact

: <sip:1060@

callbridgefqdn.any.com

```

;transport=tcp>
2018-10-12 09:09:24.510 Info SIP trace:

Ms-Conversation-ID

: 3P5Hu8grR1GGDF1BSMZAmw==
2018-10-12 09:09:24.510 Info SIP trace: To: <sip:stejanss@newany.com>
2018-10-12 09:09:24.510 Info SIP trace:

From

: "EX60 Steven" <sip:1060@
callbridgefqdn.any.com
>;tag=fb4ae780677e9d9b

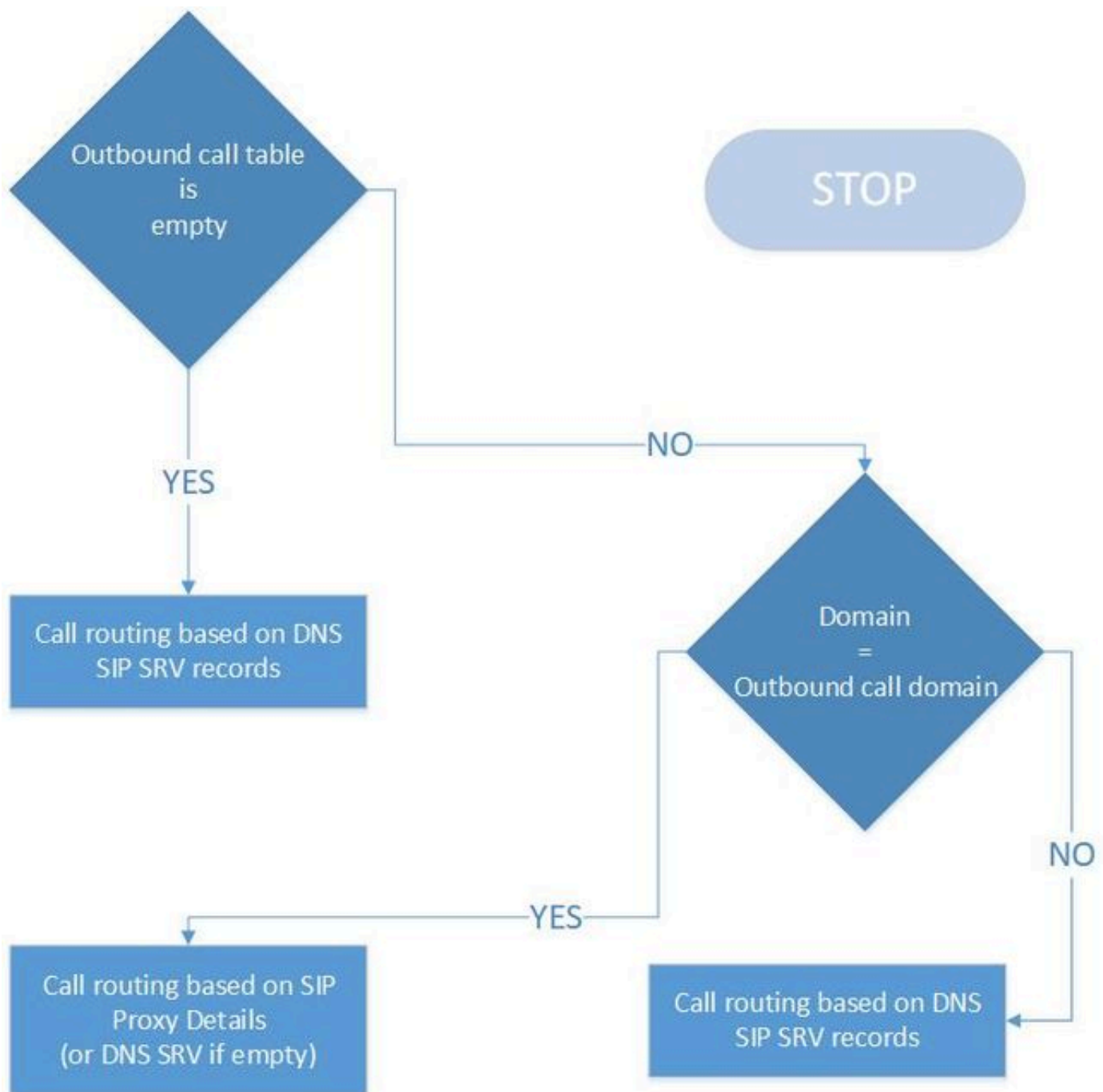
```

In case the forwarding rule would just be set at **pass through**, then there would not be any modification on the From header as seen as well from the previous example (in which case pass through was set on the forwarding rule). The Contact header is always adapted as CMS starts a new callLeg and thus must add in a Contact header to itself.

Different combinations of **Caller ID** and **Local Contact Domain** and **Local From Domain** can be used. The From header on the outbound SIP INVITE is constructed as shown on the table where the inbound call enters the CMS with a From header of `usera@from.com`.

| Forwarding rule Caller ID | Outbound call rule Local contact domain | Outbound call rule Local from domain | Resulting from header |
|---------------------------|---|--------------------------------------|-------------------------|
| Pass through | NA | NA | usera@from.com |
| Use dial plan | NA | <u>newfrom.com</u> | usera@newfrom.com |
| Use dial plan | cms1.test.cms.com | <blank> | usera@cms1.test.cms.com |
| Use dial plan | <blank> | <blank> | usera@<ip_cms> |

Step 3. Outbound Call Table



This is the last table in the call routing logic that makes the call out to a different server as:

- The incoming call is not handled locally (on the incoming call matching domain).
- It an outbound call from a CMS space (via CMA or via API in case of TMS scheduled meetings for example or Cisco Meeting Manager (CMM) instructed outbound call) or from a CMA client.

From the image, you can see that the logic is relatively easy. If there is no entry at all in the table, it still allows for outbound calls but assumes that the CMS server is able to resolve on SIP SRV records (`_sips._tcp` / `_sip._tcp` / `_sip._udp`) for that particular domain as mentioned on the SIP Request URI. If the table is not empty, but there is no match for the dialed domain then the same DNS lookup logic is performed. If there is a match on the domain, then it follows on the logic of that particular rule. In that regards, if you want to block outbound calls from CMA or as made via TMS or CMM, you can do this in two ways. Either do not have any DNS SRV records (or not resolvable by CMS) or route those calls to your call control (CUCM or Expressway for example) and block the calls there.

The image shows an example outbound call table:

Outbound calls

| Filter | Domain | SIP proxy to use | Local contact domain | Local from domain | Trunk type | Behavior | Priority | Encryption |
|--------------------------|----------------------|-----------------------|------------------------|----------------------------|--------------|----------|----------|-------------|
| <input type="checkbox"/> | steven.lab | <none; call directly> | contact.test.com | test.com | Standard SIP | Stop | 5 | Unencrypted |
| <input type="checkbox"/> | newany.com | 10.48.36.46 | callbridgefqdn.any.com | callbridgefqdn.any.com | Lync | Stop | 4 | Unencrypted |
| <input type="checkbox"/> | any.com | 10.48.36.46 | | <use local contact domain> | Standard SIP | Stop | 3 | Unencrypted |
| <input type="checkbox"/> | test.cms.com | 10.48.36.46 | | <use local contact domain> | Standard SIP | Stop | 2 | Unencrypted |
| <input type="checkbox"/> | vcs.steven.lab | 10.48.36.46 | | <use local contact domain> | Standard SIP | Stop | 1 | Unencrypted |
| <input type="checkbox"/> | <match all domains> | 10.48.36.215 | | <use local contact domain> | Standard SIP | Stop | 0 | Unencrypted |
| | <input type="text"/> | <input type="text"/> | <input type="text"/> | <input type="text"/> | Standard SIP | Stop | 0 | Auto |

With a general **<match all domains>** rule at the end and the first rule to the domain of steven.lab without a **SIP Proxy to use** filled in (so it relies on DNS SRV records for it).

Note that this is an ordered list with a higher **priority** value that is covered first. In case you match a rule with the **Behavior** set to Stop, the call does not go through the rest of the table after that match and the call has failed if that SIP Proxy failed to route the call for example. When that setting would be set to Continue, you could allow for a fallback to a different route or different node in the cluster. For example, you can specify a different SIP proxy for each rule to the same domain.

The settings of **Local Contact Domain** and **Local From Domain** are covered on the previous section of the incoming call forwarding table. The **Trunk type** allows you to specify which type of call needs to be made, which can either be Standard SIP, Lync or Avaya which depends on the receiving system.

The **Encryption** field determines whether the signaling of the call must be unencrypted or encrypted. However, note that this does not imply any media encryption as that is set on the **SIP media encryption** configuration as found on the **Configuration > Call Settings** menu. On this config, you also have the option to select Auto which tries to make the call first with an encrypted signaling with a possible fallback to an unencrypted signaling. If you know up front that the other side is encrypted or unencrypted, it is highly recommended to define it accordingly to avoid any call setup delays due to the fallback process.

An example output of the log file for a call to steven.lab (after rewrite of the domain on the incoming call forwarding table), with DNS trace and SIP trace set to detailed shows us the SRV records that are queried and the fallback mechanism in case the Encryption is set to Auto.

<#root>

```
2018-10-12 11:25:16.168 Info call 821: incoming SIP call from "sip:1060@steven.lab" to local URI
2018-10-12 11:25:16.179 Info forwarding call to 'sip:stejanss@any.com' to 'stejanss@steven.lab'
2018-10-12 11:25:16.180 Info call 822:
```

outgoing SIP call

to "stejanss@

steven.lab

"

```
2018-10-12 11:25:16.180 Info DNS trace: resolving "
```

steven.lab

" (SRV "

`_sips._tcp`

"", dnsType:1) for call 822

2018-10-12 11:25:16.181 Info DNS trace: resolution of "steven.lab" (SRV "_sips._tcp") for call 822
2018-10-12 11:25:16.181 Info DNS trace: resolution of "steven.lab" (SRV "_sips._tcp") for call 822

succeeded

; results: 1

2018-10-12 11:25:16.181 Info DNS trace: resolution of "steven.lab" (SRV "_sips._tcp") for call 822

10.48.36.215:5061

2018-10-12 11:25:16.181 Info SIP trace: connection 45: allocated for outgoing encrypted connection

2018-10-12 11:25:16.201 Info

handshake error

336151576 on outgoing connection 45 to 10.48.36.215:5061 from 10.48.80.71:54864

2018-10-12 11:25:16.201 Info SIP trace: connection 45: shutting down...

2018-10-12 11:25:16.201 Info call 822:

falling back to unencrypted control connection

...

2018-10-12 11:25:16.201 Info DNS trace: resolving "steven.lab" (SRV "

`_sip._tcp`

"", dnsType:1) for call 822

2018-10-12 11:25:16.202 Info DNS trace: resolution of "steven.lab" (SRV "_sip._tcp") for call 822

2018-10-12 11:25:16.202 Info DNS trace: resolution of "steven.lab" (SRV "_sip._tcp") for call 822

succeeded

; results: 1

2018-10-12 11:25:16.202 Info DNS trace: resolution of "steven.lab" (SRV "_sip._tcp") for call 822


10.48.36.215:5060

2018-10-12 11:25:16.202 Info SIP trace: connection 46: allocated for outgoing connection to 10.48

2018-10-12 11:25:16.203 Info SIP trace: connection 46: outgoing connection successful, 10.48.80.7

2018-10-12 11:25:16.205 Info SIP trace: connection 46: outgoing SIP TCP data to 10.48.36.215:5060

2018-10-12 11:25:16.205 Info SIP trace: INVITE sip:stejanss@steven.lab SIP/2.0

 **Note:** In case of a clustered environment with multiple callbridges, you could set up outbound dialplan rules per callbridge when you configure it via API and specify on the API object a callbridge ID (or callbridgeGroup ID). Assume for example that you want all calls to go out from one particular callbridge for a particular domain (for example when you dial to us.example.com you would like it to go out from your US based servers). Then ensure that you have an API configuration for the outboundDialPlanRules so that each other callbridge than the US based one is able to route the call to the US callbridge (in case of this example).

OutboundDialPlanRule (for US callbridge)

- domain = us.example.com
- sipProxy = <empty when using DNS SRV / IP or FQDN if manually set>
- scope = callbridge
- callbridge = <UScallbridge-ID>

OutboundDialPlanRules (for all non-US callbridges that must allow to make that call) (need one per callbridge)

- domain = us.example.com
- sipProxy = <IP-or-FQDN-of-US-Callbridge>
- scope = callbridge
- callbridge = <non-US-callbridge-ID>

Verify

There is currently no verification procedure available for this configuration.

Troubleshoot

There is currently no specific *troubleshooting* information available for this configuration.

Related Information

- [Technical Support & Documentation - Cisco Systems](#)
- [Collaboration Solutions Analyzer tool](#)
- [CMS documentation](#)

NOTE: For configuration examples please consult these guides:

- [Configure and Integrate CMS Single Combined Guide](#)
- [Configure Cisco Meeting Server and CUCM Guide](#)