EDUCRONIX

TELECOMM SKILLS AT YOUR HANDS

Exploring Load Balance capabilities with

SIParator® / Firewall®

For the Ingate SIParator using software release 6.2.1 or later

Ingate SIParator®/Firewall®

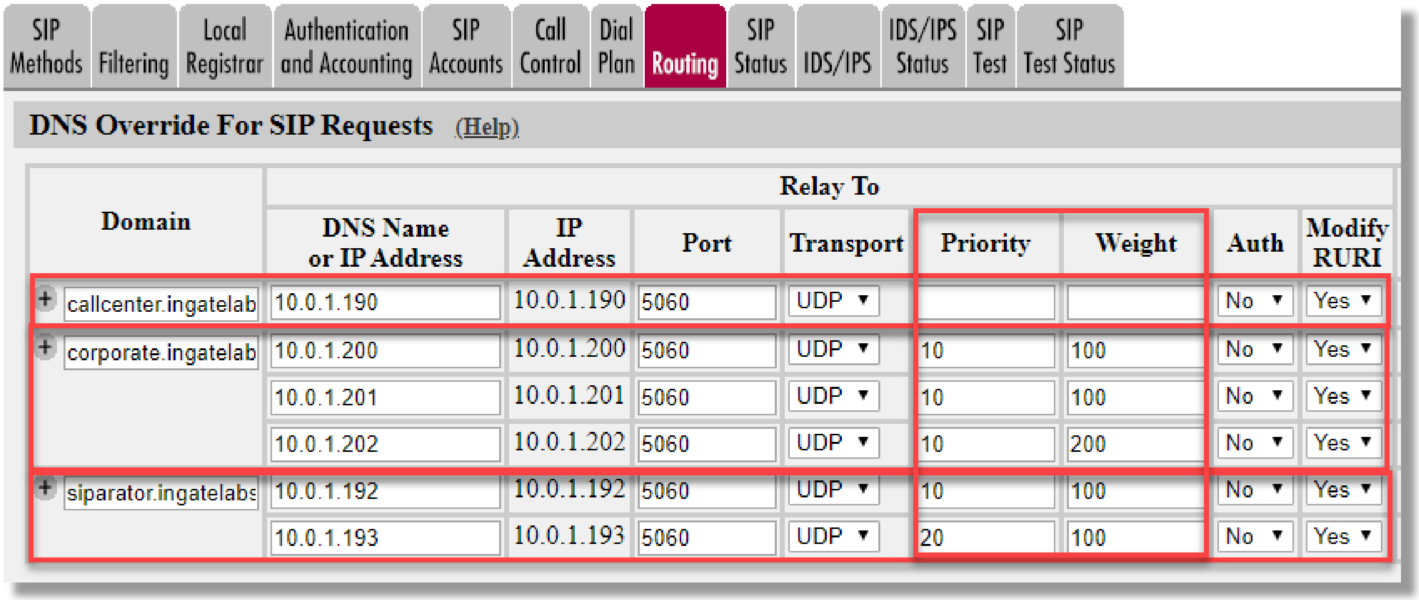
Table of Contents

1 Ingate SIParator®/Firewall® DNS Override

One of the most interesting call flow control features in SIParator®/Firewall®, besides what is already advanced in the Dial Plan, REST API Call Control, Routing and Trunk Groups, is what we call DNS Override, that can be found as a section inside **SIP Traffic→Routing**

DNS Override allows to use the SIParator®/Firewall® as a DNS Server, able to manage same functionalities of records A and SRV. This is a powerful capability as you don't need to involve existing DNS Servers when the need is to convert an FQDN/DNS Name into a destination public or private IP address.

DNS Override looks like this:



DNS Override For SIP Requests (Help)

Domain	Relay To							
	DNS Name or IP Address	IP Address	Port	Transport	Priority	Weight	Auth	Modify RURI
callcenter.ingatelab	10.0.1.190	10.0.1.190	5060	UDP			No	Yes
corporate.ingatelab	10.0.1.200	10.0.1.200	5060	UDP	10	100	No	Yes
	10.0.1.201	10.0.1.201	5060	UDP	10	100	No	Yes
	10.0.1.202	10.0.1.202	5060	UDP	10	200	No	Yes
siparator.ingatelabs	10.0.1.192	10.0.1.192	5060	UDP	10	100	No	Yes
	10.0.1.193	10.0.1.193	5060	UDP	20	100	No	Yes

Here, you enter SIP domains not handled by the firewall and which cannot be looked up using DNS.

Auth means the requestor will be asked for authentication before the SIP request is forwarded to the destination.

Modify RURI means the Request-URI of the SIP request will be rewritten with the new destination before it is forwarded.

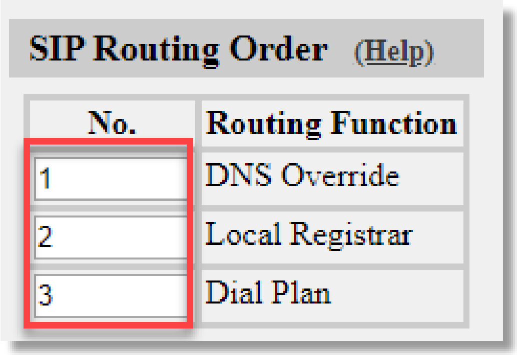
In this example we have 3 different cases:

- "callcenter…." Is a single domain resolution like an "A" record in a DNS
- "corporate…" is a load balance with uniform distribution of the domain among 3 different destinations (Like an SRV Record in a DNS)
- "siparator…." Is a failover configuration that will send all traffic to the lowest priority, and only if it is not reachable will be sent to the next priority. (Like an SRV Record in a DNS)

It is important to note that Domains used in the DNS Override table May or may not be an FQDN, so for instance if we use just a name like "loadbalancer", which can't be resolved by a DNS Server, will be valid for SIParator and will hence resolve on the IP address defined in the corresponding row. This will be helpful when explaining use cases later in this technical note.

Additionally, it is important to understand what the sequence is followed by SIParator when deciding how to route a SIP request.

Under the same SIP **Traffic → Routing** section, you'll find a "SIP Routing Order" table:



SIP Routing Order (Help)

No.	Routing Function
1	DNS Override
2	Local Registrar
3	Dial Plan

You can select in which order the different routing functions of SIParator/Firewall should be processed. Change the number to move the row up or down.

DNS Override is simply the DNS Override we already explained.

Local Registrar means all locally registered users, and all static registrations made on this section. It does not mean registration requests, only requests for the registered users.

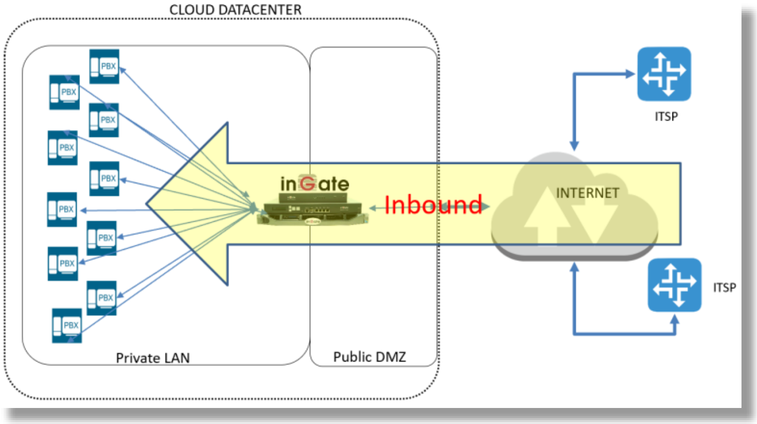
Dial Plan is simply the Dial Plan table on the Dial Plan page.

2 Load Balancing Use Cases.

For the purpose of this document we will distinguish 3 common cases:

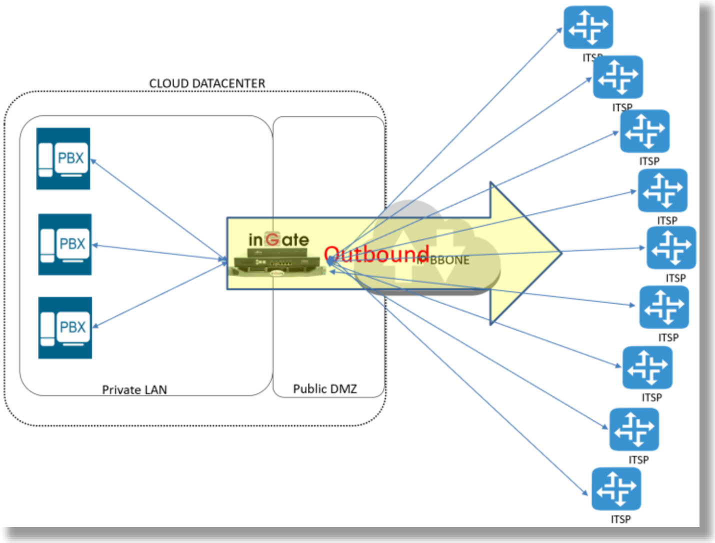
2.1 SIP Trunk Inbound Load Balance

In this use case we want to distribute among several SIP destinations inbound SIP traffic coming from one or more trunks/ITSP. Distribution Policies will be based on priorities and weights depending on the case needs.



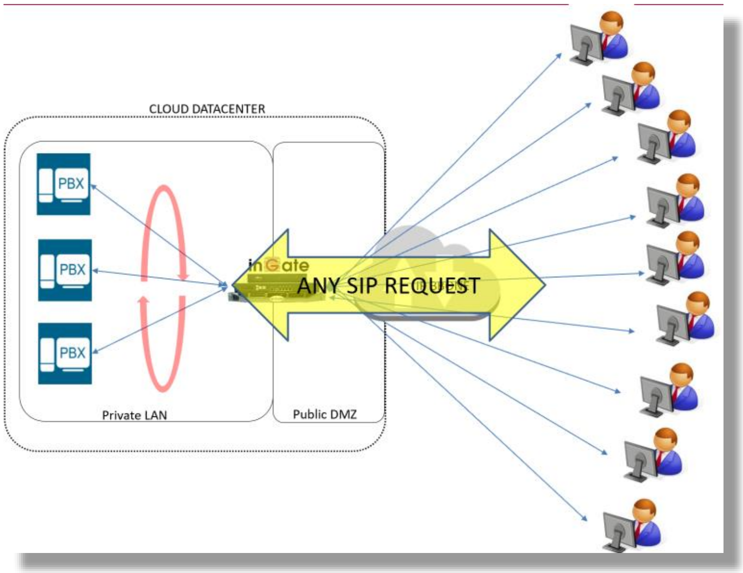
2.2 SIP Trunk Outbound Load Balance

In this use case we want to distribute among several Trunks/ITSP coming from one or more SIP platforms (IP-PBX, UC, Call Center, Dialer, etc..). Distribution Policies will be based on priorities and weights depending on the case needs.



2.3 Remote endpoints/users Load Balance

In this case, we are using SIParator®/Firewall® as the front end for any remote user/endpoint looking to register and use a Proxy Server (IPPBX, UC, etc.…) sitting behind the SBC. All SIP requests will be load balanced among several platforms, even for balancing traffic or having a failover strategy.

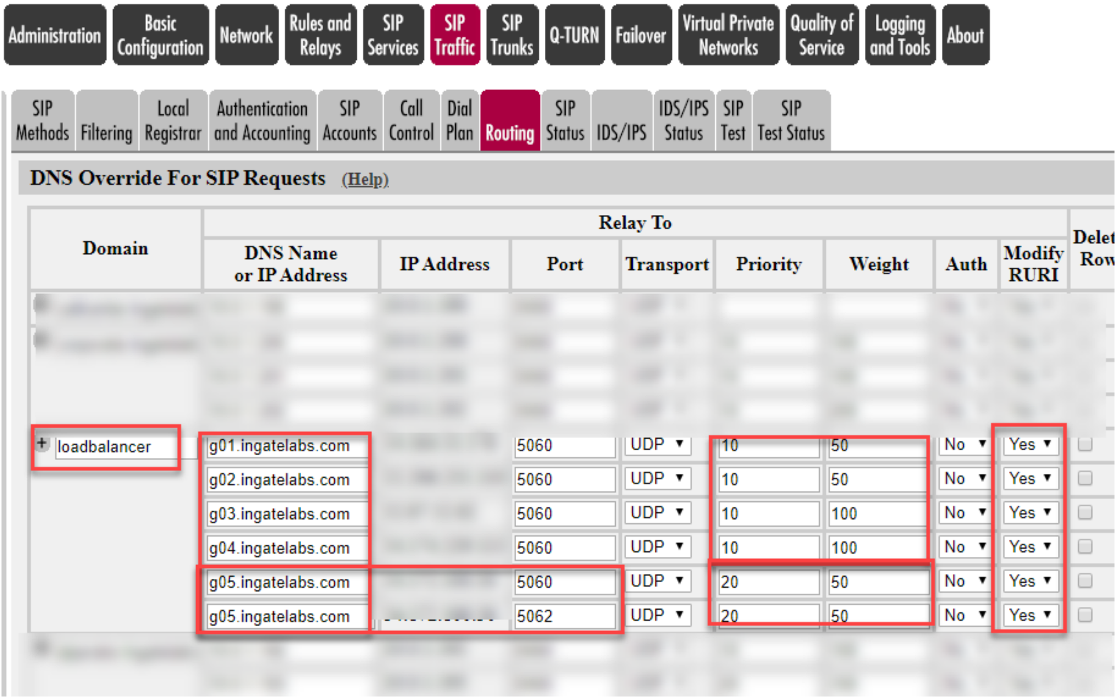


3 Load Balancing explained using DNS Override with Ingate SIParator®/Firewall®

Before we go in-depth with each use case, we will build a generic load balancer inside the SIParator®/Firewall® to have a better understanding on how it works.

As previously said in section 1, we will take advantage of the fact that de domain column in the DNS Override table doesn't need to be an FQDN, so we will use a generic name that will not resolve any IP by other DNS services.

This is an example that help us explain how DNS Override works when load balance or failover is needed:



DNS Override For SIP Requests (Help)

Domain	Relay To								Delet Row
	DNS Name or IP Address	IP Address	Port	Transport	Priority	Weight	Auth	Modify RURI	
[redacted]									
+ loadbalancer	g01.ingatelabs.com	[redacted]	5060	UDP ▾	10	50	No ▾	Yes ▾	☐
	g02.ingatelabs.com	[redacted]	5060	UDP ▾	10	50	No ▾	Yes ▾	☐
	g03.ingatelabs.com	[redacted]	5060	UDP ▾	10	100	No ▾	Yes ▾	☐
	g04.ingatelabs.com	[redacted]	5060	UDP ▾	10	100	No ▾	Yes ▾	☐
	g05.ingatelabs.com	[redacted]	5060	UDP ▾	20	50	No ▾	Yes ▾	☐
	g05.ingatelabs.com	[redacted]	5062	UDP ▾	20	50	No ▾	Yes ▾	☐
[redacted]									

Here we created a domain name "loadbalancer", which will distribute calls with the following strategy:

- Destinations g01, g02, g03, and g04 with same priority will carry calls with such weights that let g01 and g02 with 1 third of the traffic (evenly), and g03 and g04 getting 1 third each.
- There will be 2 trunk destination on g05 (port 5060 and 5062) and they will take all the traffic shared 50/50 in case g01, g02, g03 and g04 are not reachable.
- G05 is a fail over destination for all the balanced group made of g01, g02, g03 and g04.
- In case we need R-URI to be build using the DNS name or IP address shown in that column we will select "yes" in the Modify R_URI column.

In summary, this load balancer table distributes call received among g01 to g05 destinations 1/6 of the traffic to g01, 1/6 to g02, 1/3 to g03 and 1/3 to g04. g05 will take full load in case the other 4 become unreachable and will receive 50/50 distributed in 2 ports.

Next steps will show, on each use case how to take advantage of this feature.

4 RFC 3263 Considerations when building DNS SRV weighted records.

If we review section 4.4 in RFC 3263 we will find the following recommendations:

- To make processing easier for stateless proxies, it is RECOMMENDED that domain administrators make the weights of SRV records with equal priority different (for example, using weights of 1000 and 1001 if two servers are equivalent, rather than assigning both a weight of 1000), and similarly for NAPTR records.

- If the weights are the same (e.g. 1000) they are sorted on the target name instead of the weights.

Better to have high weight values in series (i.e. 10000, 10001, 10002, etc.)
Otherwise you will get a distortion if you follow the algorithm in RFC 2782.

Having explained that from this point on we will use high numbers and never equal. A very small difference will not affect the statistical result.

5 Load Balance outbound traffic (trunk)

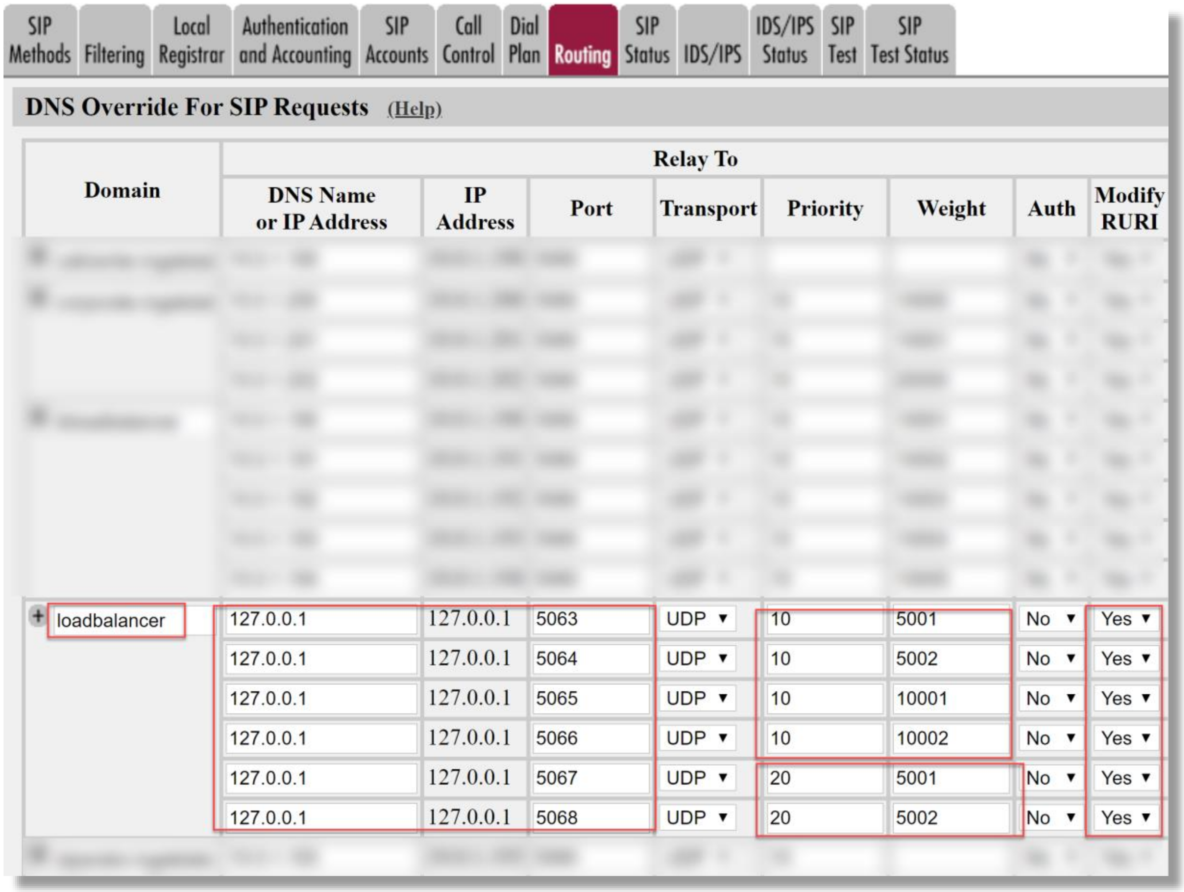
For outbound load balancer we are assuming we have several trunks and we need to distribute calls among them based on certain policy.

There are two potential options to solve this case, the first one is by using Trunk Groups to connect to each one of the SIP Trunk destinations among which we will distribute the calls. The second option is by implementing all call control without using trunk groups and doing all in the Dial Plan.

5.1 Using Trunk Groups

5.1.1 DNS Override

When using Trunk Groups, the Load balancer will use Ports to discriminate each one of the possible trunk destinations. For instance, if, based on the load balancer, a call needs to be sent to the second trunk in the possible destination, the call will be sent to por 5064.



DNS Override For SIP Requests (Help)

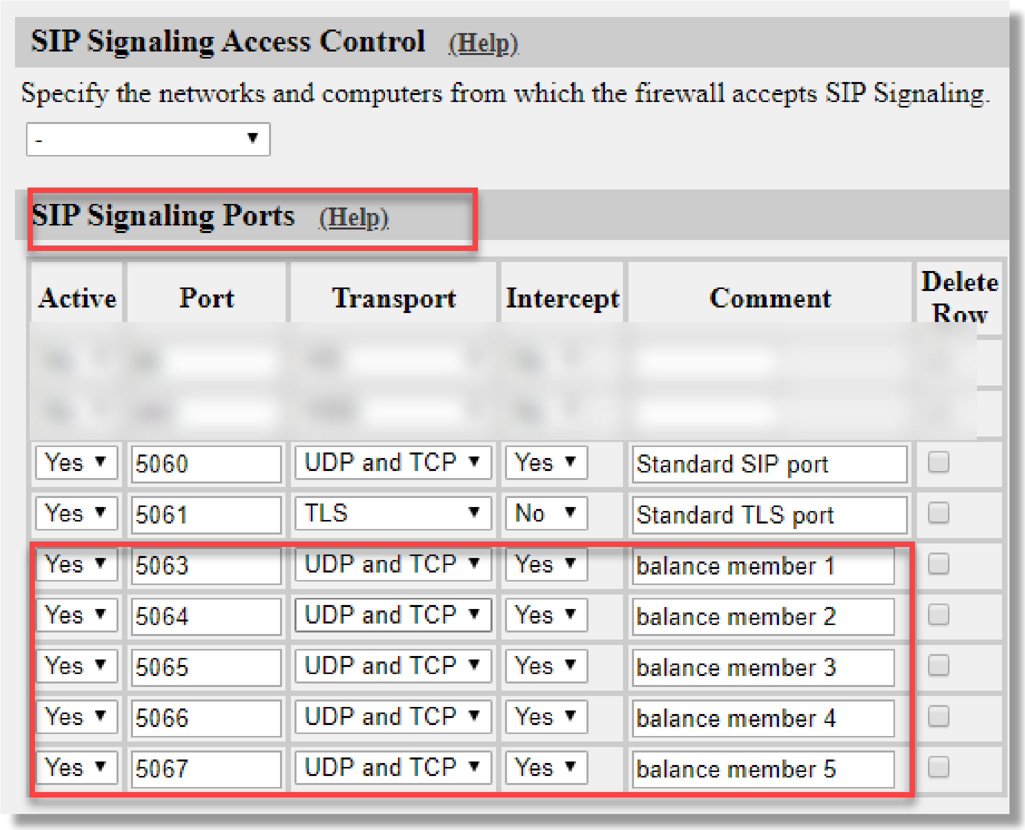
Domain	Relay To							
	DNS Name or IP Address	IP Address	Port	Transport	Priority	Weight	Auth	Modify RURI
[redacted]								
loadbalancer	127.0.0.1	127.0.0.1	5063	UDP ▾	10	5001	No ▾	Yes ▾
	127.0.0.1	127.0.0.1	5064	UDP ▾	10	5002	No ▾	Yes ▾
	127.0.0.1	127.0.0.1	5065	UDP ▾	10	10001	No ▾	Yes ▾
	127.0.0.1	127.0.0.1	5066	UDP ▾	10	10002	No ▾	Yes ▾
	127.0.0.1	127.0.0.1	5067	UDP ▾	20	5001	No ▾	Yes ▾
	127.0.0.1	127.0.0.1	5068	UDP ▾	20	5002	No ▾	Yes ▾
[redacted]								

In this example, we are uniformly distributing calls in the following criteria:

- 1/3rd of the calls are evenly distributed between first and second trunk (5062 and 5064)
- 1/3rd of the calls will be send to third trunk (5065)
- 1/3rd of the calls will be send to the fourth trunk (5066)
- Port 5067 and 5068 will be used only for failover ("20">"10" → higher priority) and will distribute evenly to each one of them.

5.1.2 Enable SIP Ports

In order to enable SIParator to listen on all the ports we have defined to use in the DNS Override, we will need to add them in the SIP Configuration:



SIP Signaling Access Control (Help)

Specify the networks and computers from which the firewall accepts SIP Signaling.

- ▾

SIP Signaling Ports (Help)

Active	Port	Transport	Intercept	Comment	Delete Row
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
Yes ▾	5060	UDP and TCP ▾	Yes ▾	Standard SIP port	□
Yes ▾	5061	TLS ▾	No ▾	Standard TLS port	□
Yes ▾	5063	UDP and TCP ▾	Yes ▾	balance member 1	□
Yes ▾	5064	UDP and TCP ▾	Yes ▾	balance member 2	□
Yes ▾	5065	UDP and TCP ▾	Yes ▾	balance member 3	□
Yes ▾	5066	UDP and TCP ▾	Yes ▾	balance member 4	□
Yes ▾	5067	UDP and TCP ▾	Yes ▾	balance member 5	□

5.1.3 Dial Plan Matching criteria (phase 1)

In order to intercept outbound traffic (from IPPBX to PSTN) and send it to the load balancer ("loadbalancer") from the Dial Plan, we will use matching criteria (From Header and Request-uri).

Let's say for instance, in our case we have three IP PBX that will be sending outbound traffic to the SIParator, and ten the SIParator will need to load balance among five trunks with the strategy we already built in our "loadbalancer".

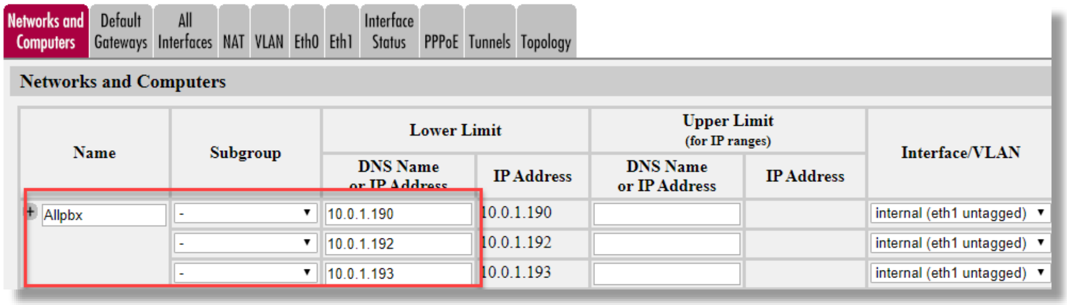
The three PBX are located at:

10.0.1.190

10.0.1.192

10.0.1.193

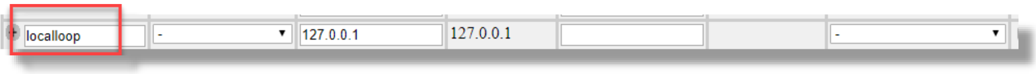
We will create a "Network and Computers" name to include those 3 PBX:



Networks and Computers

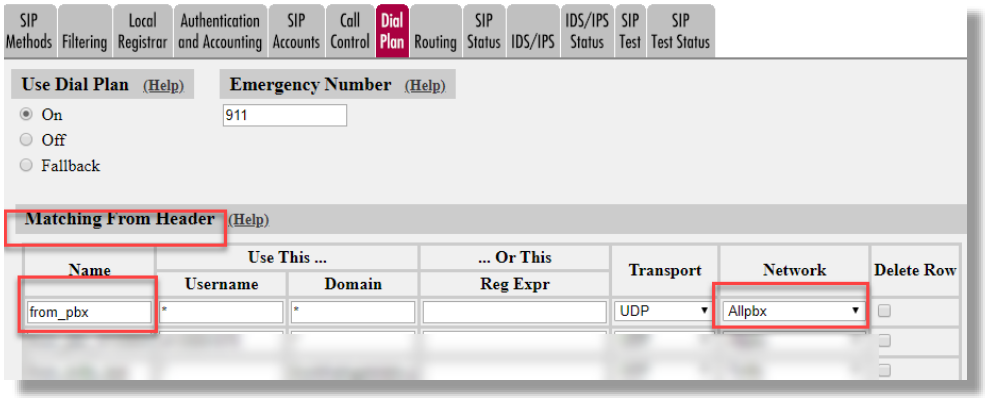
Name	Subgroup	Lower Limit		Upper Limit (for IP ranges)		Interface/VLAN
		DNS Name or IP Address	IP Address	DNS Name or IP Address	IP Address	
Allpbx	- ▾	10.0.1.190	10.0.1.190			internal (eth1 untagged) ▾
	- ▾	10.0.1.192	10.0.1.192			internal (eth1 untagged) ▾
	- ▾	10.0.1.193	10.0.1.193			internal (eth1 untagged) ▾

We will also use a name to refer to the ingate itself, let's call it "localloop"



localloop	- ▼	127.0.0.1	127.0.0.1			- ▼

In the Dial plan, we'll match the "from header" with any of the IP PBX IP addresses:



Use Dial Plan (Help)

● On
○ Off
○ Fallback

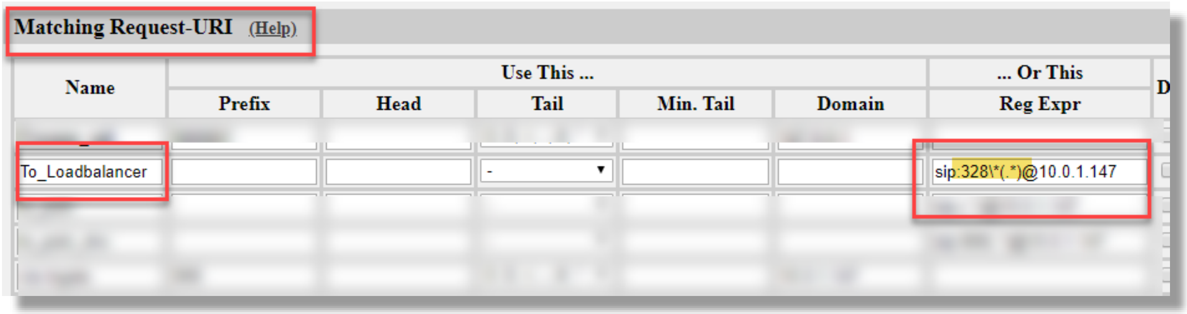
Emergency Number (Help)

911

Matching From Header (Help)

Name	Use This ...		... Or This	Transport	Network	Delete Row
	Username	Domain	Reg Expr			
from_pbx	*	*		UDP ▼	Allpbx ▼	□

Also, as PBX's may have different criteria to decide if a call needs to be balanced or goes to a different destination, we will assume for the example here that all calls from any IPPBX will have a 328* prefix for any outbound call that needs to be managed with the load balancer.



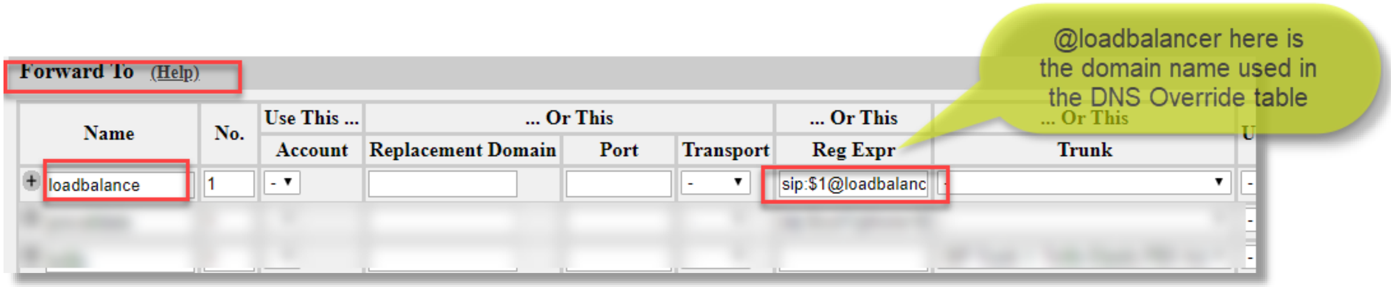
Matching Request-URI (Help)

Name	Use This ...					... Or This	D
	Prefix	Head	Tail	Min. Tail	Domain	Reg Expr	
To_Loadbalancer			- ▼			sip:328*(.*)@10.0.1.147	□

This match will catch calls arriving to SIParator (10.0.1.147) with the prefix 328* and will strip it off, leaving the real destination in the captured digits in "(.*)".

5.1.4 Forward to balancer and Dial Plan Phase 1.

Then we create the route (Forward to) destination to be able to send the load balance call to the load balancer.

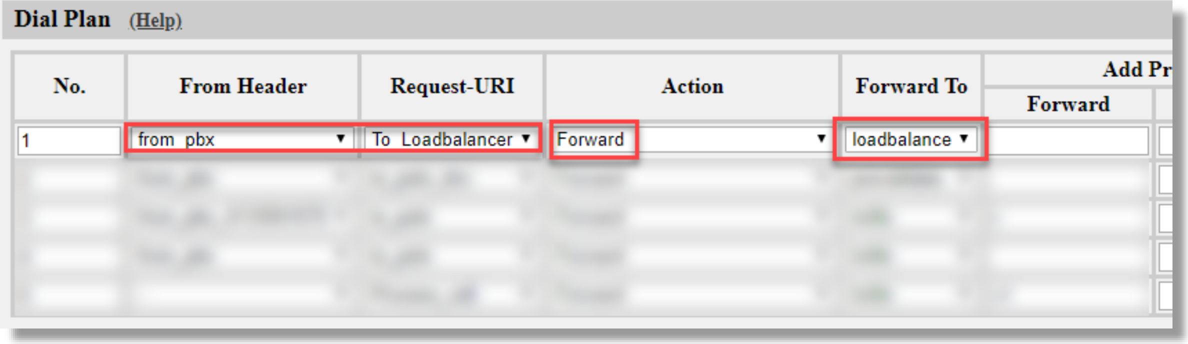


Forward To (Help)

Name	No.	Use This ...	... Or This			... Or This	... Or This	U
		Account	Replacement Domain	Port	Transport	Reg Expr	Trunk	
loadbalance	1	- ▼			- ▼	sip:$1@loadbalanc	- ▼	-

@loadbalancer here is the domain name used in the DNS Override table

Finally, the actual dial plan will route the call to the 'loadbalancer" if the From Header and Request-URI match the criteria explained above.



Dial Plan (Help)

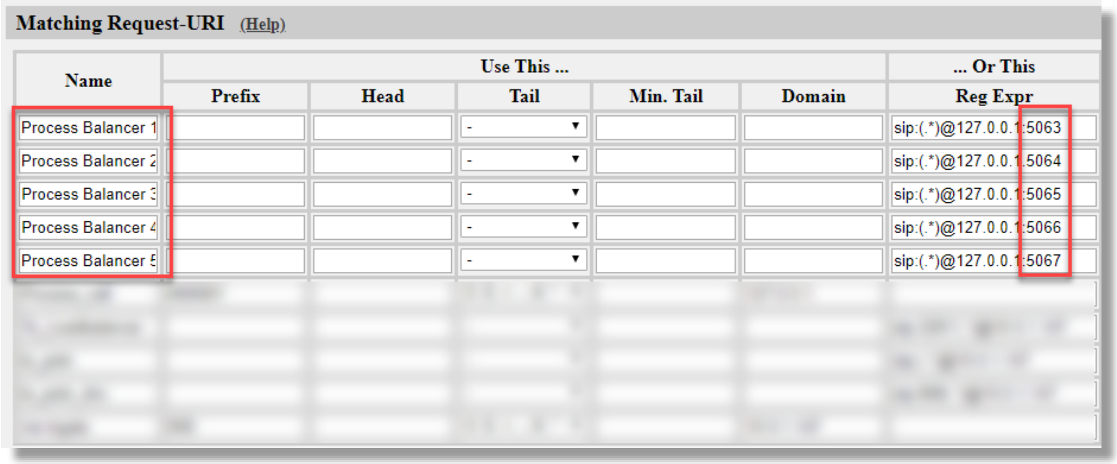
No.	From Header	Request-URI	Action		Forward To	Add Pr Forward
1	from pbx ▾	To Loadbalancer ▾	Forward	▾	loadbalance ▾	

5.1.5 Phase 2 Dial Plan

As the match and route by this dial plan, it will be sent to a domain name "loadbalancer", this will land on the DNS Override table to resolve such name and then the load balance strategy will be applied. Calls will pass second time thru the Dial Plan, and now we will filter by the destination port used by DNS Override.

5.1.6 Matching Phase 2:

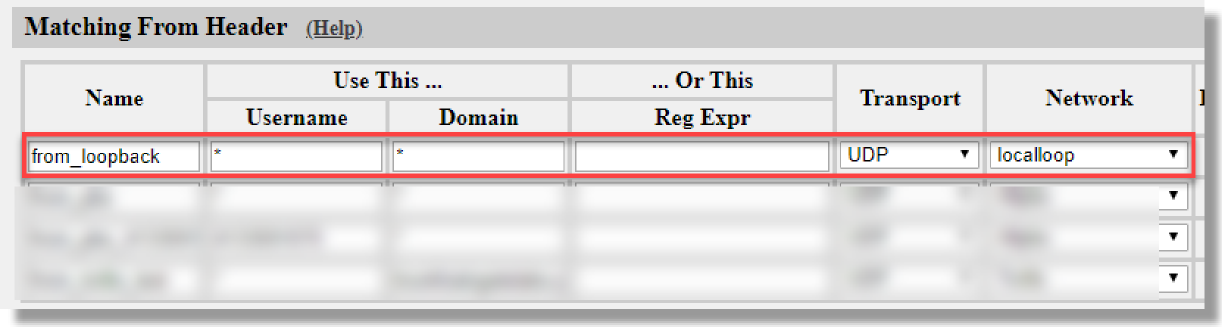
To match the second pass will add in the matching Request URI the port matching criteria:



Matching Request-URI (Help)

Name	Use This ...					... Or This
	Prefix	Head	Tail	Min. Tail	Domain	Reg Expr
Process Balancer 1			- ▾			sip:(.*)@127.0.0.1:5063
Process Balancer 2			- ▾			sip:(.*)@127.0.0.1:5064
Process Balancer 3			- ▾			sip:(.*)@127.0.0.1:5065
Process Balancer 4			- ▾			sip:(.*)@127.0.0.1:5066
Process Balancer 5			- ▾			sip:(.*)@127.0.0.1:5067

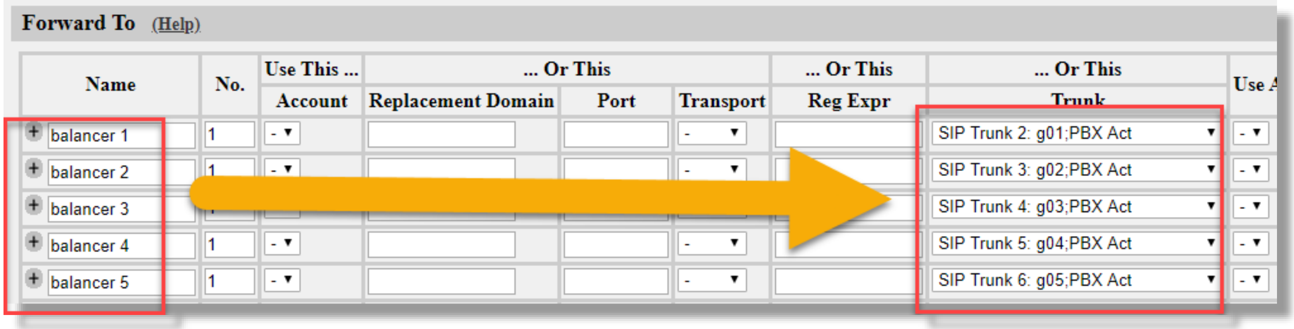
We will also match the From Header to validate the call is coming from the local loop interface:



Matching From Header (Help)

Name	Use This ...		... Or This	Transport	Network
	Username	Domain	Reg Expr		
from_loopback	*	*		UDP	localloop
[illegible]				[illegible]	[illegible]
[illegible]				[illegible]	[illegible]
[illegible]		[illegible]		[illegible]	[illegible]

Now, we need to add all trunks as potential destinations (Forward to) based on the previous matches:

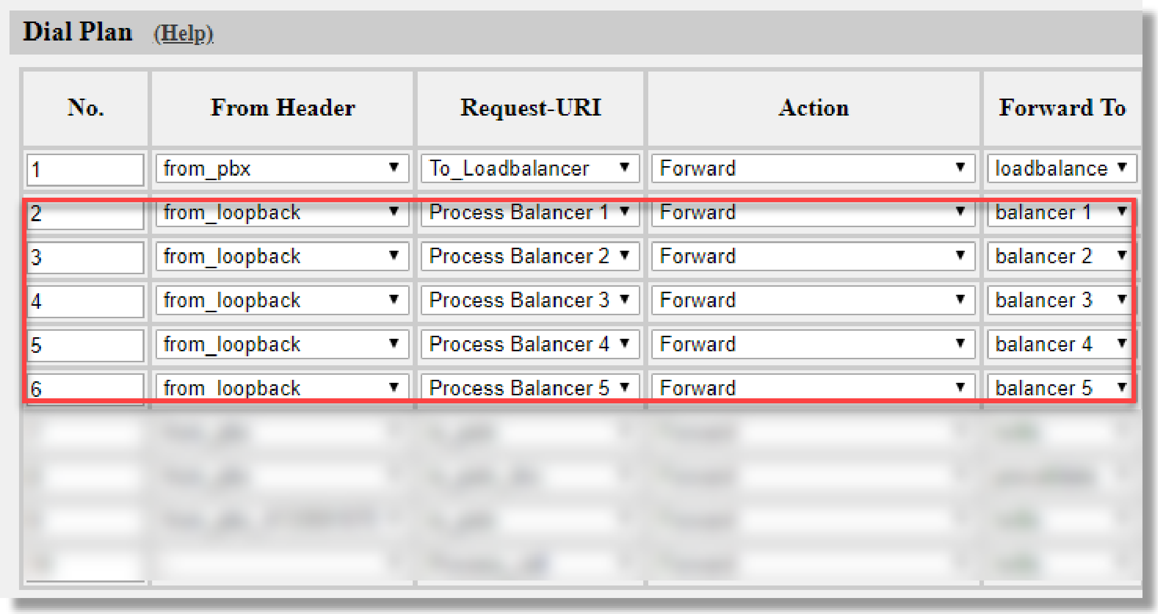


Forward To (Help)

Name	No.	Use This ...	... Or This			... Or This	... Or This	Use A
		Account	Replacement Domain	Port	Transport	Reg Expr	Trunk	
balancer 1	1	-			-		SIP Trunk 2: g01;PBX Act	-
balancer 2	1	-			-		SIP Trunk 3: g02;PBX Act	-
balancer 3	[illegible]	[illegible]					SIP Trunk 4: g03;PBX Act	-
balancer 4	1	-			-		SIP Trunk 5: g04;PBX Act	-
balancer 5	1	-			-		SIP Trunk 6: g05;PBX Act	-

5.1.7 Dial Plan Phase 2

Finally, the dial plan should look like this:



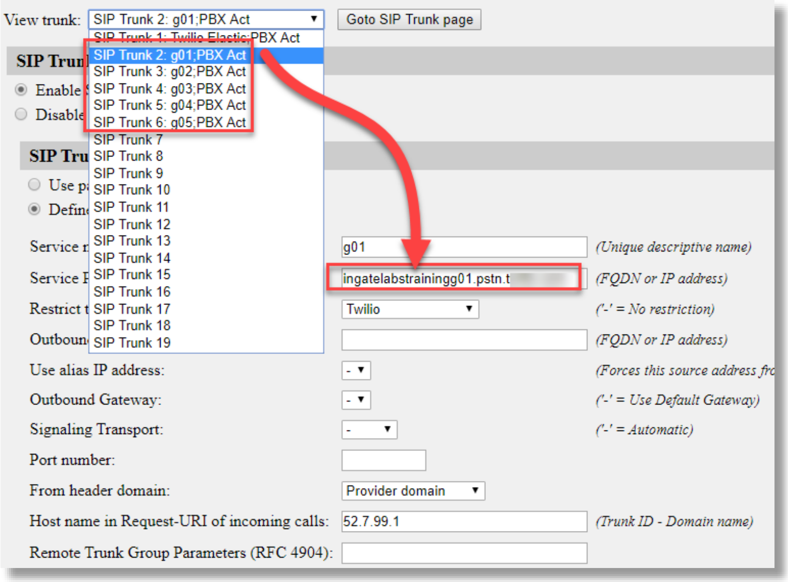
Dial Plan (Help)

No.	From Header	Request-URI	Action	Forward To
1	from_pbx	To_Loadbalancer	Forward	loadbalance
2	from_loopback	Process Balancer 1	Forward	balancer 1
3	from_loopback	Process Balancer 2	Forward	balancer 2
4	from_loopback	Process Balancer 3	Forward	balancer 3
5	from_loopback	Process Balancer 4	Forward	balancer 4
6	from_loopback	Process Balancer 5	Forward	balancer 5
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]		[illegible]	[illegible]	[illegible]

5.1.8 SIP Trunk Group Definition

Each one of the Trunk Groups will be configured depending on the ITSP' used. In our example we illustrate using "Twilio"

Each Trunk will look like this:



View trunk: SIP Trunk 2: g01;PBX Act ▼ [Goto SIP Trunk page]

SIP Trunk 1: Twilio Elastic;PBX Act
SIP Trunk 2: g01;PBX Act
SIP Trunk 3: g02;PBX Act
SIP Trunk 4: g03;PBX Act
SIP Trunk 5: g04;PBX Act
SIP Trunk 6: g05;PBX Act
SIP Trunk 7
SIP Trunk 8
SIP Trunk 9
SIP Trunk 10
SIP Trunk 11
SIP Trunk 12
SIP Trunk 13
SIP Trunk 14
SIP Trunk 15
SIP Trunk 16
SIP Trunk 17
SIP Trunk 18
SIP Trunk 19

SIP Trun

● Enable
○ Disable

SIP Tru

○ Use p
● Defin

Service r: g01 *(Unique descriptive name)*

Service F: ingatelabstrainingg01.pstn.t[redacted] *(FQDN or IP address)*

Restrict t: Twilio ▼ *('-' = No restriction)*

Outboun: *(FQDN or IP address)*

Use alias IP address: - ▼ *(Forces this source address fro*

Outbound Gateway: - ▼ *('-' = Use Default Gateway)*

Signaling Transport: - ▼ *('-' = Automatic)*

Port number:

From header domain: Provider domain ▼

Host name in Request-URI of incoming calls: 52.7.99.1 *(Trunk ID - Domain name)*

Remote Trunk Group Parameters (RFC 4904):

…… At this point everything is ready to test the balancer!!!

5.2 Using plain Dial Plan.

This case is recommended only if you have a clear understanding of all interoperability issues that normally are resolved in the SIP Trunk Groups and any additional manipulation needed to make te connection happen.

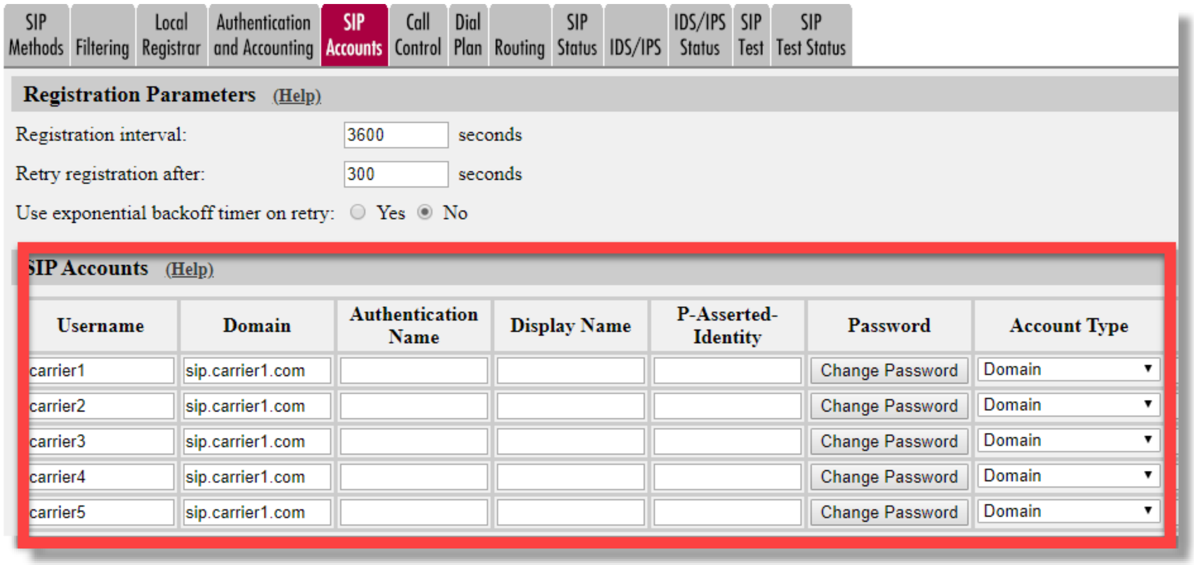
We can use the same strategy to distribute among all the SIP destinations using Local loop with ports associated to each destination

You can follow the same steps we did in the previous section except for the Dial Plan which will be slightly different and bypass all the SIP Trunk Group definition.

The difference will be in the Forward to Section where rather that sending the calls to the corresponding Trunk Group, will send the call to specific sip destination using any of the other exclusive options:

5.2.1 Created SIP account.

Using SIP Traffic→SIP Accounts, you can define SIP Destination to route traffic:



Registration Parameters (Help)

Registration interval: 3600 seconds

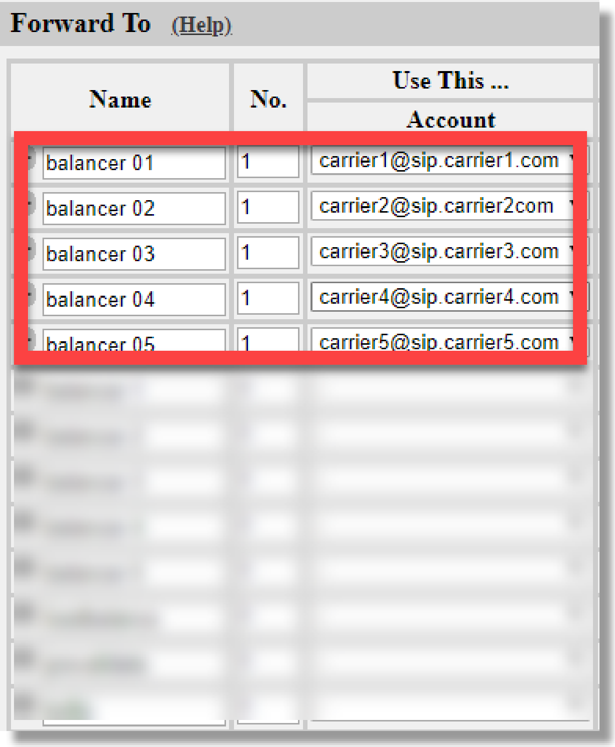
Retry registration after: 300 seconds

Use exponential backoff timer on retry: ○ Yes ● No

SIP Accounts (Help)

Username	Domain	Authentication Name	Display Name	P-Asserted-Identity	Password	Account Type
carrier1	sip.carrier1.com				Change Password	Domain
carrier2	sip.carrier1.com				Change Password	Domain
carrier3	sip.carrier1.com				Change Password	Domain
carrier4	sip.carrier1.com				Change Password	Domain
carrier5	sip.carrier1.com				Change Password	Domain

Then we can refer to those destination in the “Forward to” in our dial plan:

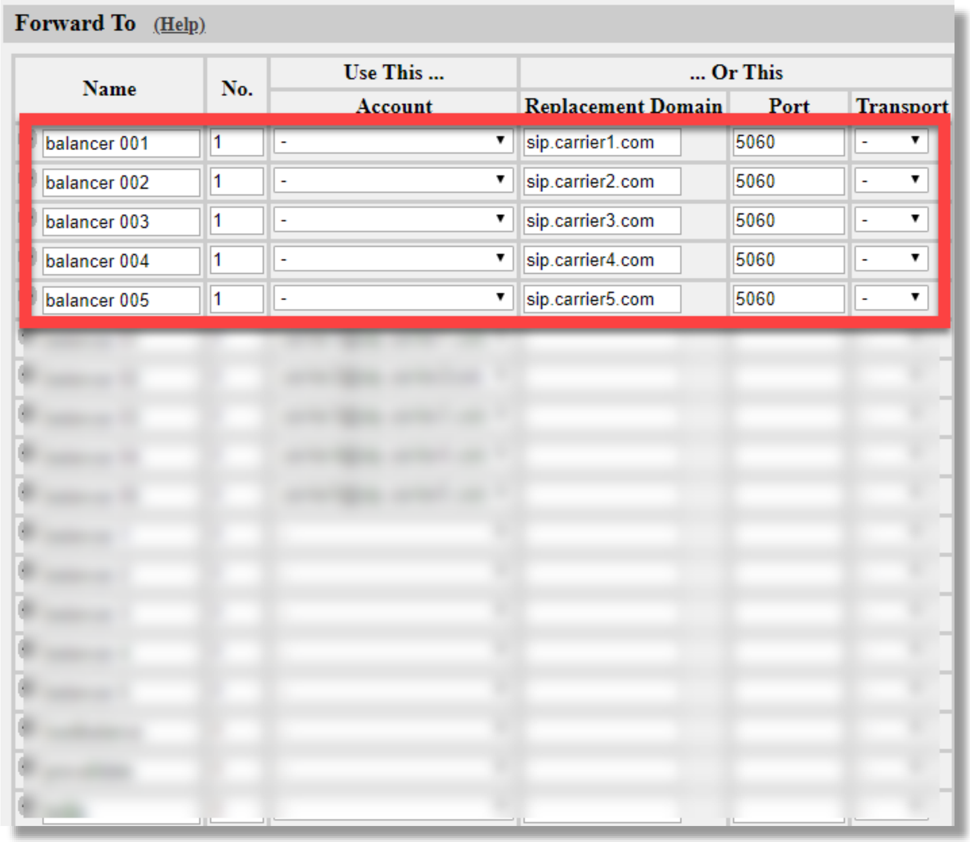


Forward To (Help)

Name	No.	Use This ...
		Account
balancer 01	1	carrier1@sip.carrier1.com
balancer 02	1	carrier2@sip.carrier2com
balancer 03	1	carrier3@sip.carrier3.com
balancer 04	1	carrier4@sip.carrier4.com
balancer 05	1	carrier5@sip.carrier5.com

5.2.2 Domain Replacement:

In this case, alternatively we can use the Domain Replacement Section in the “forward to” section



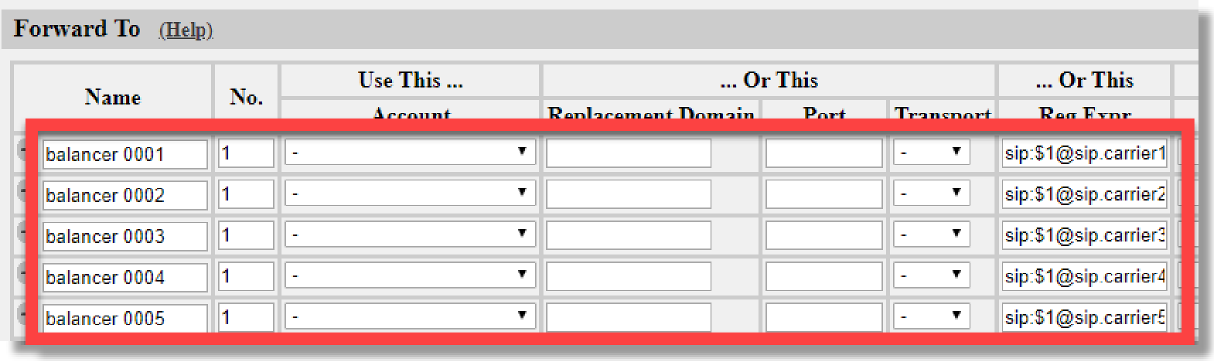
Forward To (Help)

Name	No.	Use This ...	... Or This		
		Account	Replacement Domain	Port	Transport
balancer 001	1	-	sip.carrier1.com	5060	-
balancer 002	1	-	sip.carrier2.com	5060	-
balancer 003	1	-	sip.carrier3.com	5060	-
balancer 004	1	-	sip.carrier4.com	5060	-
balancer 005	1	-	sip.carrier5.com	5060	-

5.2.3 Using Regular Expressions.

Here we can use full potential of Ingate's GHM (Generic Header Manipulation) and regular expression to build the destination of any trunk. For detailed information and discovery of all the potential review this document: https://www.ingate.com/appnotes/How_To_use_Generic_Header_Manipulation.pdf

This is a simplified example:

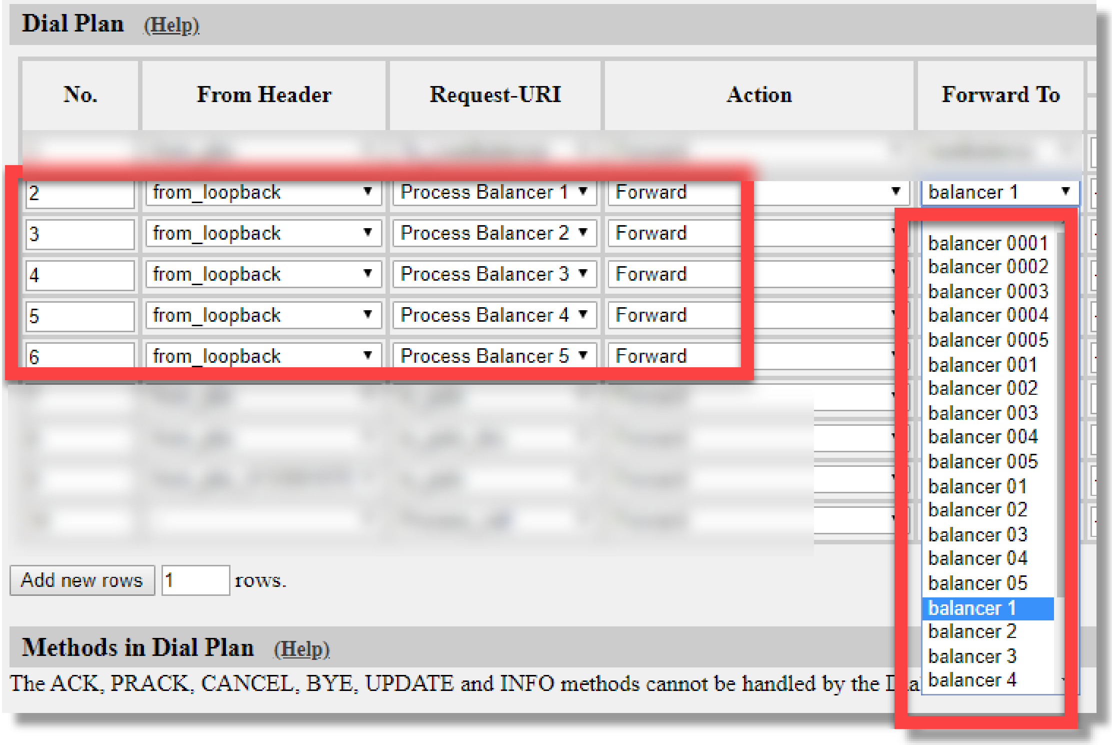


Forward To (Help)

Name	No.	Use This ...	... Or This			... Or This
		Account	Replacement Domain	Port	Transport	Reg Expr
balancer 0001	1	-			-	sip:$1@sip.carrier1
balancer 0002	1	-			-	sip:$1@sip.carrier2
balancer 0003	1	-			-	sip:$1@sip.carrier3
balancer 0004	1	-			-	sip:$1@sip.carrier4
balancer 0005	1	-			-	sip:$1@sip.carrier5

5.2.4 Final dial plan

After selecting which option will be used in the "Forward to", we will need to adjust the Dial Plan table with the correspondent destinations.



Dial Plan (Help)

No.	From Header	Request-URI	Action	Forward To
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
2	from_loopback	Process Balancer 1	Forward	balancer 1
3	from_loopback	Process Balancer 2	Forward	
4	from_loopback	Process Balancer 3	Forward	
5	from_loopback	Process Balancer 4	Forward	
6	from_loopback	Process Balancer 5	Forward	
[redacted]	[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	[redacted]	

1 rows.

Methods in Dial Plan (Help)

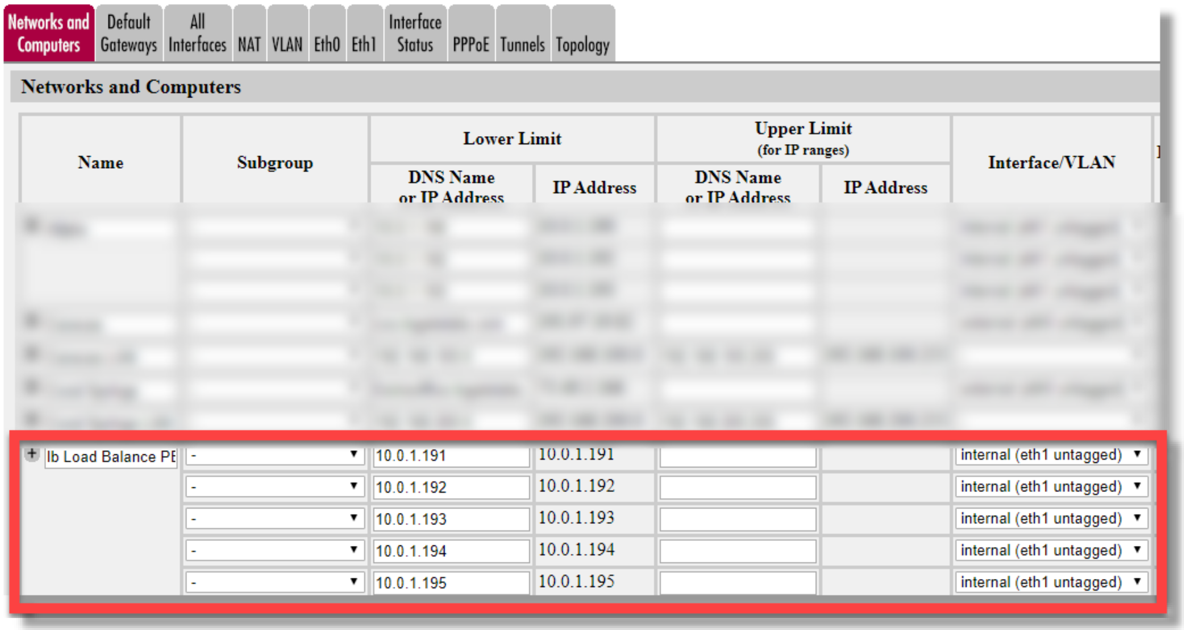
The ACK, PRACK, CANCEL, BYE, UPDATE and INFO methods cannot be handled by the D

6 Load Balance Inbound Traffic (trunk)

In this section we will explain how to load balance SIP traffic coming from PSTN via SIP Trunk Groups, and terminated in several Voip Platforms (IPPBX, UC, etc.…).

It is not far of the cases we presented in previous section, but now the distribution point will be the PBX section in the Trunk Group definition, where the PBS destination will be the load balancer and this one, defined in the DNS Override, will define the load balance strategy among all the IP destinations.

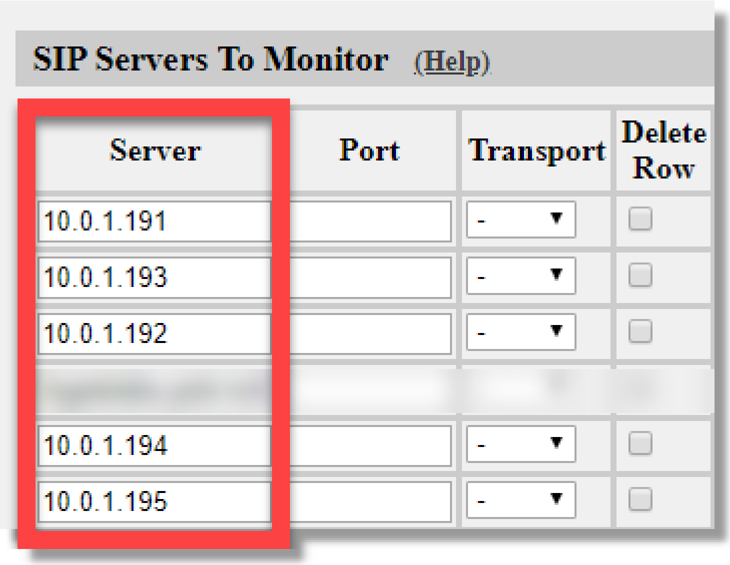
Lets first create a network name for all the IP's that participate as a member of the load balancing group destinations.



Networks and Computers

Name	Subgroup	Lower Limit		Upper Limit (for IP ranges)		Interface/VLAN
		DNS Name or IP Address	IP Address	DNS Name or IP Address	IP Address	
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
lb Load Balance PE	- ▼	10.0.1.191	10.0.1.191			internal (eth1 untagged) ▼
	- ▼	10.0.1.192	10.0.1.192			internal (eth1 untagged) ▼
	- ▼	10.0.1.193	10.0.1.193			internal (eth1 untagged) ▼
	- ▼	10.0.1.194	10.0.1.194			internal (eth1 untagged) ▼
	- ▼	10.0.1.195	10.0.1.195			internal (eth1 untagged) ▼

It is always suggested to add those destinations to the SIP Monitor under SIP Services.



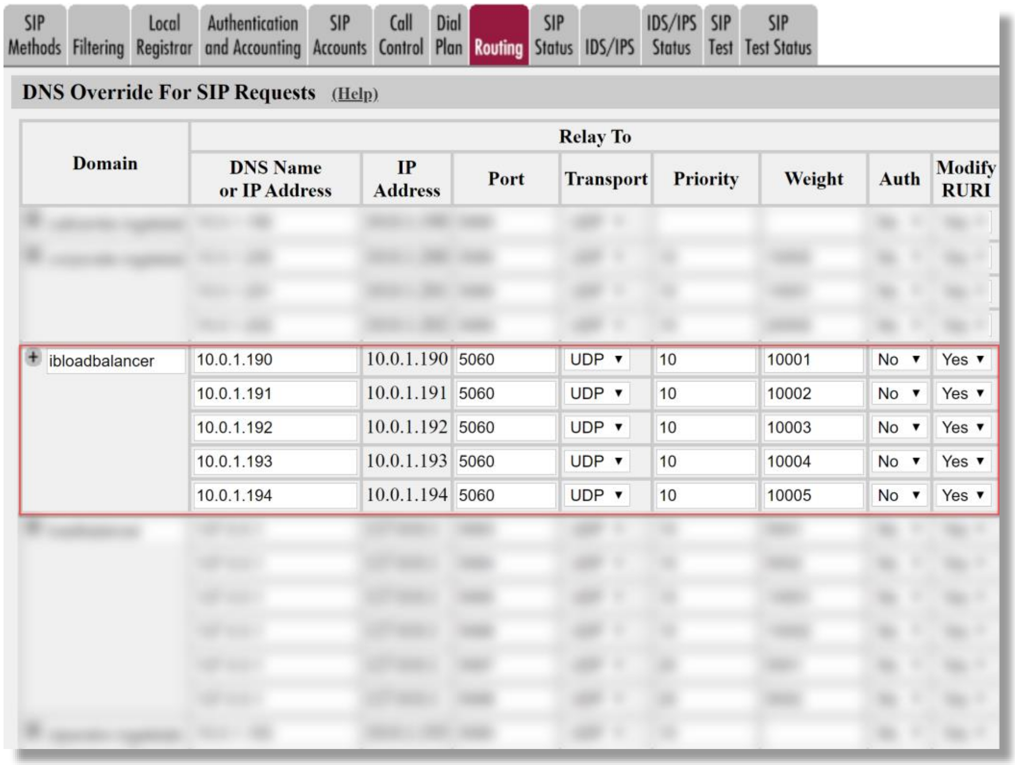
SIP Servers To Monitor (Help)

Server	Port	Transport	Delete Row
10.0.1.191		- ▼	☐
10.0.1.193		- ▼	☐
10.0.1.192		- ▼	☐
[redacted]	[redacted]	[redacted]	[redacted]
10.0.1.194		- ▼	☐
10.0.1.195		- ▼	☐

6.1 Define the Load Balancer destination in the Trunk Group.

Any inbound call coming thru a SIP Trunk Page, and after matching all the rules defined there, will be sent to the PBX defined at the bottom of the Trunk Group.

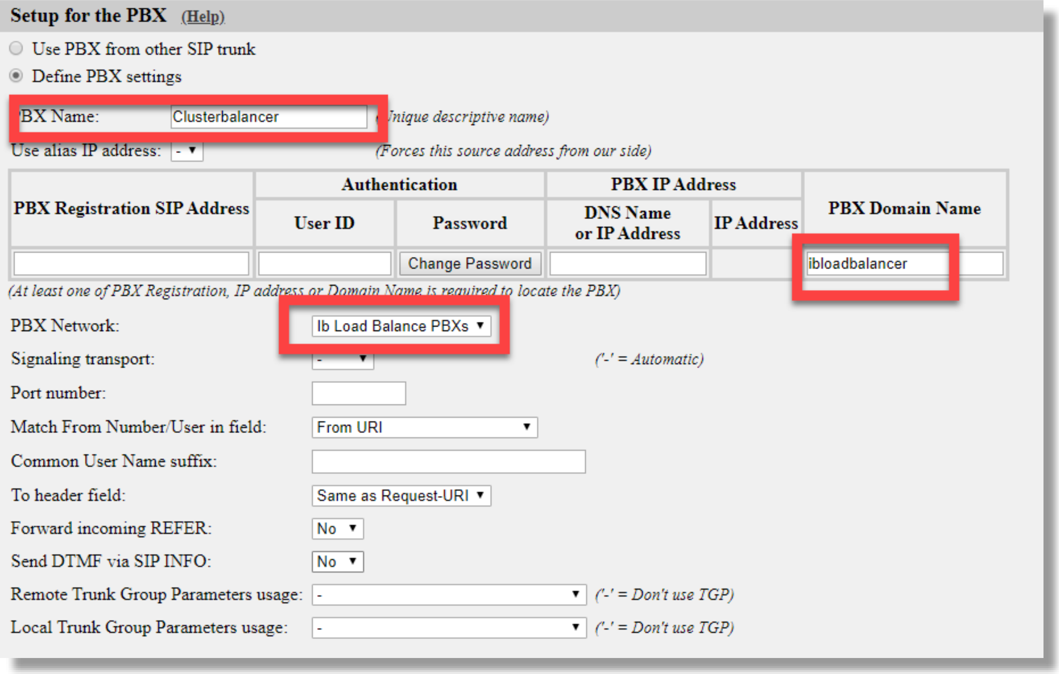
Here is where we will use a local domain (the name of the load balance domain defined in the DNS Override) shown here:



DNS Override For SIP Requests (Help)

Domain	Relay To							
	DNS Name or IP Address	IP Address	Port	Transport	Priority	Weight	Auth	Modify RURI
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]			[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
ibloadbalancer	10.0.1.190	10.0.1.190	5060	UDP	10	10001	No	Yes
	10.0.1.191	10.0.1.191	5060	UDP	10	10002	No	Yes
	10.0.1.192	10.0.1.192	5060	UDP	10	10003	No	Yes
	10.0.1.193	10.0.1.193	5060	UDP	10	10004	No	Yes
	10.0.1.194	10.0.1.194	5060	UDP	10	10005	No	Yes
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]			[illegible]	[illegible]

Once this has been defined, we can add a new PBX destination that will send inbound call to the balancer.



Setup for the PBX (Help)

○ Use PBX from other SIP trunk

● Define PBX settings

PBX Name: Clusterbalancer (Unique descriptive name)

Use alias IP address: - (Forces this source address from our side)

PBX Registration SIP Address	Authentication		PBX IP Address		PBX Domain Name
	User ID	Password	DNS Name or IP Address	IP Address	
		Change Password			ibloadbalancer

(At least one of PBX Registration, IP address or Domain Name is required to locate the PBX)

PBX Network: Ib Load Balance PBXs

Signaling transport: - ('-' = Automatic)

Port number:

Match From Number/User in field: From URI

Common User Name suffix:

To header field: Same as Request-URI

Forward incoming REFER: No

Send DTMF via SIP INFO: No

Remote Trunk Group Parameters usage: - ('-' = Don't use TGP)

Local Trunk Group Parameters usage: - ('-' = Don't use TGP)

We should define this PBX destination, with a designated name, in order to use it in other Trunk Groups. This will allow for having several trunks, sharing the same balancer as a destination.

PBX Network will be limited to the IP's address group name we defined previously.

The PBX domain will be the domain name we just added in the DNS Override table.

6.2 Extended capabilities.

As we are using Trunk Groups, and they provide the capability to define N-to-N flows, we can be very selective on when a call will be sent to the balance for instance just looking at the DID.

It is important to emphasize that the use of DNS Override, not only allows for Weighted load balance traffic, but al to define failover strategies.

7 Load Balance for remote users/endpoints.

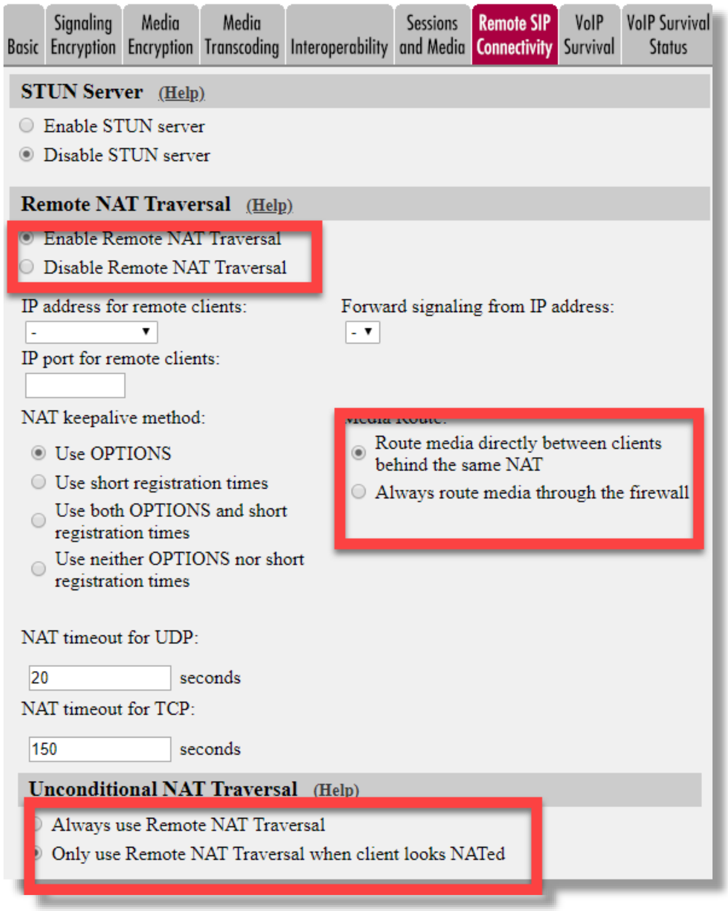
Remote user access to SIP infrastructure is the second most used application on SBC's.

SIParator/Firewall from Ingate has been designed to easily integrate and secure access for such remote users or endpoints.

For more details about the use case see this documentation: https://www.ingate.com/files/AWS/Ingate-SIParator-Configuration-Guide-Secure-Voip-for-Remote-Users-on-AWS.pdf

There are two sections needed for implementing remote users:

- SIP Services → Remote SIP Connectivity



STUN Server (Help)

○ Enable STUN server
● Disable STUN server

Remote NAT Traversal (Help)

● Enable Remote NAT Traversal
○ Disable Remote NAT Traversal

IP address for remote clients: -

Forward signaling from IP address: -

IP port for remote clients:

NAT keepalive method:

● Use OPTIONS
○ Use short registration times
○ Use both OPTIONS and short registration times
○ Use neither OPTIONS nor short registration times

Media Route:

● Route media directly between clients behind the same NAT
○ Always route media through the firewall

NAT timeout for UDP: 20 seconds

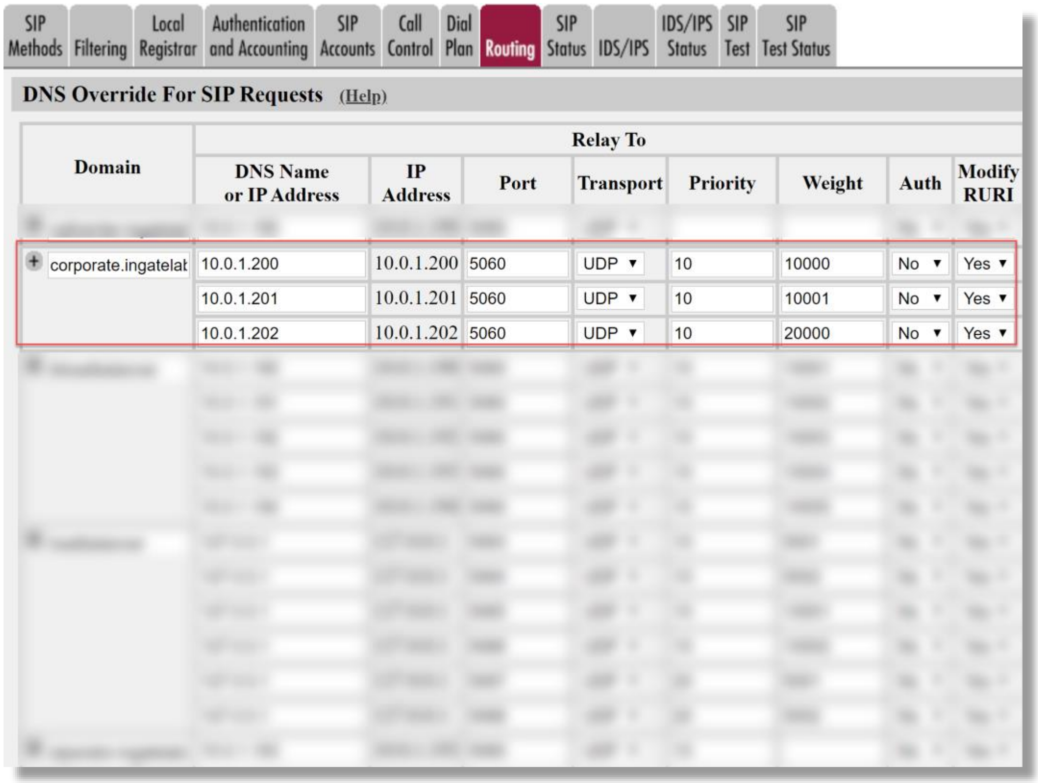
NAT timeout for TCP: 150 seconds

Unconditional NAT Traversal (Help)

Always use Remote NAT Traversal
Only use Remote NAT Traversal when client looks NATed

- SIP Traffic → Routing

  In this case we will use again the DNS Override, but now the domain to use will be the one the remote users or endpoints use as registrar domain.



DNS Override For SIP Requests (Help)

Domain	Relay To							
	DNS Name or IP Address	IP Address	Port	Transport	Priority	Weight	Auth	Modify RURI
corporate.ingatelat	10.0.1.200	10.0.1.200	5060	UDP	10	10000	No	Yes
	10.0.1.201	10.0.1.201	5060	UDP	10	10001	No	Yes
	10.0.1.202	10.0.1.202	5060	UDP	10	20000	No	Yes

  Here you can define any strategy that better fits you needs, including load/weighted balance, failover, etc.

8 Final comments and summary.

Siparator/Firewall from Ingate, besides all the known and rich features, in security, interoperability, and media path enhancements, is also a powerful and feature rich, flexible and easy to use SIP Load Balancer.

All functionalities include typical DNS SRV alike Records.

It can be used for inbound or outbound traffic distribution as well as Remote access SIP services.