Application Note



Advanced Call Control using HTTP Services and RestAPI

with Ingate SIParator[®] SBC

Introduction	3
About Ingate SIParator [®] SBC product family	3
SIParator [®] Call Control	4
RestAPI Architecture	6
The RestAPI loopback model with SIParator [®]	7
Deployment scenarios	7
Use case description	7
Proof of Concept Topology	8
Configuring SIParator [®] SBC	10
Pre-requisites	10
Configuring Interface	11
Other Network related configurations	13
Configuring SIP in SIParator [®]	14
Setup SIP Ports	14
Setup SIP Filtering	14
Setup SIP Monitoring	15
Setup Call Control for RestAPI	16
Configure HTTP Services to provide RestAPI Server behavior	17
Configure Dial Plan	22
How easy is to maintain this Configuration?	27
Semi-manual update	27
Automated Updates option	29
Disclaimers	
Help and Support	

Introduction

About Ingate SIParator® SBC product family.

A Session Border Controller is a device that connects to an existing network firewall to seamlessly enable SIP communications (Session Initiation Protocol). While traditional firewalls block SIP traffic – including mission-critical applications like Voice over IP (VoIP) – the Ingate SIParator[®] SBC resolves this problem, working in tandem with your current security solutions.

The Ingate SIParator[®] is a powerful, flexible and cost-effective Enterprise Session Border Controller (E-SBC) for SIP connectivity, security and interoperability, such as connecting PBXs and Unified Communications (UC) solutions to SIP Trunking service providers.

The Ingate Firewall[®], which is always included in the product, makes the Ingate SIParator an all-in-one appliance for data security as well as session border control.



Ingate's SIParators[®]/Firewalls[®] are available in a range of models:

The SIParator simplifies SIP trunking and makes it easy to connect remote UC end points, aggregate SIP trunks and distribute sessions between sites and service delivery points. It's utilized for Real-Time communications security, SIP interoperability and extensive connectivity. The SIParator[®] is compatible with all existing networks and comes with a standard SIP proxy and a SIP registrar. It has support for NAT

and PAT as well as for TLS and SRTP to encrypt both SIP signaling and media, eliminating the security issue most associated with using enterprise VoIP.

The flexible system of add-on licenses allows any enterprise to enhance the SIParator[®]/Firewall[®] solution to meet their needs at any given moment.

With more than 10,000 installations worldwide, the Ingate SIParator[®] comes in a wide range of capacities, and has been used by retail companies, financial institutions, industrial firms, government agencies, call centers and small-to-large enterprises.

SIParator® Call Control

Since version 6.2 SIParator added a new functionality to enable RestAPI client and be able to execute HTTP Requests to a Web Service during call setup to take routing decisions in real time.

Administration	Basic Configuration	Network	HTTP Services	SIP Services	SIP Traffic	SIP Trunks	Q-TURN	V
Methods Filter	Local ring Registrar	Authentic	ation Acc	ounts STI	Call Contro	Dial Plan	Routing	Ac
REST API	Servers (H	ielp)						
ID Prefix	Suffix Reque	est Timeo	ut Cach	e Lifetim	e Delet	e Row		
Add new roo	ws 1 rov	VS.						
TLS CA C	ertificates	(<u>Help)</u>						
CA Delete	Row							
Add new row	ws 1 rov	VS.						
Logging	(<u>Help</u>)							
O Enable v	erbose loggir	g						
O Disable \	verbose loggi	ng						
Save Und	lo							

Here you can define REST API servers for call control features in the Dial Plan. We will expand later in this document about this functionality.

Lately with SIParator[®] Release 6.4 Ingate introduced HTTP Services features, which allows to use the SIParator[®] as an HTTP Proxy frontend.

	- Connocs	Services	ic Trunks	Networks	Servic	e and Tools	AUUUL	Log out				
orage and WebSockets and HTTP												
Storago Ropositorios a	nd Tunnels (H	elp)										
Enable												
) Disable												
Local Files (Utilization	n: 0.07% Space	e available:	2.0 MiB) (Help)								
A file hosted locally on the	e unit. Upload/Edit	t the file conte	nts. Add a fil	e to a File Group	below.							
Name	P	ath		File Name			File				Information	
customertable	1		custo	mers ison		Upload	Downlo	ad Edit	Size: 1057 bytes MIME type: text/plain			
									SHA256: db6743dac	91f6d92f1e53f9	191332dd54a59ae8d54afe0cc	i63b544c4ac003
zoomeytensions	1		exter	sions ison		Upload	Downlo	ad Edit	Size: 395 bytes			
Zoomextensions			CARCIN	310113.]3011		opidad	Downio		SHA256: cf0843548	82b41ee14988	83c7b176ef7e3ea1a1073acd4	461fd1366898c
Add new rows 1 row	vs.											
Local File Groups (H	<u>elp)</u>											
A group of files that are h	osted locally. A Fil	le is defined al	oove. Attach	a file group to a R	epository	that is define	d below.					
Name	Fi	le	Delete Row									
(files	customertab	le 🗸										
	zoomextensi	ions 🗸										
Add new rows 1 grou	ps with 1 row	s per group.										
Local Endpoints (Hel	9 <u>)</u>											
A local endpoint serves a	s an entry point fo	r locally and re	emotely host	ed files. It can also	serve as	an entry point	for HTT	P tunnels. A	Repository must	have a local	endpoint defined.	
Name	Destand	ID A	Ideas	Deut	Se	rver	Peer	Verification	TLS		Allow From	Delete De
Name	Protocol	IP AC	uress	Pon	Certi	ficate	Tru	sted CAs	Settin	gs	Allow From	Delete Ro
restapi	HTTP V	eth0 (10.1.0).145) 🗸	8080 -		\sim	-	\sim] [-	\sim	·	
Add new rows 1 row	VS.											
Remote Endpoint Ser	ver Groups <u>(He</u>	elp)										
A group of servers that he	st files available f	or retrieval thr	ough this un	it. Attach a server	group to a	Remote End	point the	at is defined	below.			
Name IP Address Port	Load Balance	Delete Row										
	Weight Backup											
Add new rows 1 grou	ips with 1 row	s per group.										
Remote Endpoints (lelp)											
A remote endpoint define	s how remote serv	vers should be	contacted.	Attach a remote en	dpoint to a	Repository	that is de	efined below	ν.			
Name Protocol	ver Group	Client	Peer Verit	fication TI	.S Dele	te Row						
Name	Load Balance C	ertificate Tru	isted CAs S	Server Name Sett	ings							
Add new rows 1 row	VS.											
Add new rows 1 row	VS.											
Add new rows 1 rov Methods (<u>Heip</u>)	VS.											
Add new rows 1 rov Methods (Help) A group of HTTP methods	vs. 3. Use a method g	roup in a Rep	ository that	is defined below.								
Add new rows 1 row Methods (Help) A group of HTTP methods Name	vs. s. Use a method g Metho	roup in a Rep od De	ository that lete Row	is defined below.								
Add new rows 1 row Methods (Help) A group of HTTP method: Name DEFAULT	s. Use a method g Metho GET	roup in a Rep od De	ository that lete Row	is defined below.								
Add new rows 1 row Methods (Heip) A group of HTTP method: © DEFAULT Add new rows 1 grou	s. Use a method g Metho GET ps with 1 row	roup in a Rep od De	ository that lete Row	is defined below.								
Add new rows 1 row Methods (Help) A group of HTTP methods DEFAULT Add new rows 1 grou	s. Use a method g Metho GET ps with 1 row	roup in a Rep od De S per group.	ository that lete Row	is defined below.								
Add new rows 1 row Methods (<u>Help</u>) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun	s. Use a method g Metho GET ups with 1 row	roup in a Rep od De S per group.	ository that lete Row	is defined below.								
Add new rows 1 row Methods (<u>Hein</u>) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun A repository defines stora	s. Use a method g Metho GET ups with 1 row nels (Help) ge for local and/or	roup in a Rep od De s per group.	ository that lete Row	is defined below. I/Remote Endpoi	nts and Lc	ocal File Gro	ıps abo	ve. HTTP tu	nnels via the Loca	I Endpoint c	an also be enabled here.	
Add new rows 1 row Methods (Hein) A group of HTTP method DEFAULT Add new rows 1 grow Repositories and Tun A repository defines store Name	s. Use a method g Metho GET ups with 1 row nels (Help) ge for local and/or Local	roup in a Rep od De > s per group.	ository that lete Row Define Loca Remote	is defined below. I/Remote Endpoi Allowed	nts and Lo	ocal File Gro	ips abor	ve. HTTP tu	nnels via the Loca Delete Pow	l Endpoint c	an also be enabled here.	
Add new rows 1 row Methods (Hein) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun A repository defines stora Name	S. Use a method g GET igs with 1 row nels (Help) ge for local and/or Local Endpoint	roup in a Rep od De s per group. r remote files. Local File Group	ository that lete Row Define Loca Remote Endpoint	is defined below. I/Remote Endpoi Allowed Methods	nts and Lo	ocal File Groi Ti Allow To	ıps abo ınnel	ve. HTTP tu Ports	nnels via the Loca Delete Row	l Endpoint c	an also be enabled here.	
Add new rows 1 row Methods (Heig) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun A repository defines stora Name restlocalrepository	s. Use a method g GET ups with 1 row nels (Help) ge for local and/o Local Endpoint restapi	roup in a Rep od De s per group. r remote files. Local File Group files V	ository that lete Row Define Loce Remote Endpoint	I/Remote Endpoi Allowed Methods DEFAULT	nts and Lo	ocal File Gro Tr Allow To	ips abo innel	ve. HTTP tu Ports	nnels via the Loca Delete Row	l Endpoint c	an also be enabled here.	
Add new rows 1 row Methods (Heip) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun A repository defines store Name restlocalrepository Add new rows 1 row	s. Use a method g Mething GET ups with 1 row nels (Hele) ge for local and/or Local Endpoint restapi	roup in a Rep od De s per group. r remote files. Local File Group files	Define Loca Remote Endpoint	I/Remote Endpoi Allowed Methods	nts and Lo	ocal File Gro Tr Allow To	ips abo innel	ve. HTTP tu Ports	nnels via the Loca Delete Row	l Endpoint c	an also be enabled here.	
Add new rows 1 row Methods (Help) A group of HTTP method DEFAULT Add new rows 1 grou Repositories and Tun A repository defines store Name restlocalrepository Add new rows 1 row	s. Use a method g Methu GET ups with 1 row nels (<u>Help</u>) ge for local and/o Local Endpoint [restapi]	roup in a Rep od De s per group. rremote files. Local File Group files V	Define Loca Remote Endpoint	I/Remote Endpoi Allowed Methods DEFAULT	nts and Lo	ocal File Gron Ti Allow To	ips abo innel	ve. HTTP tu Ports	Delete Row	l Endpoint c	an also be enabled here.	

We will also expand on this new feature later in this document.

Now, what if we start thinking on combining Call Control RestAPI and HTTP Service? In other words, what could be the benefit of using the RestAPI client call control and use the SIParator[®] itself also as a RestAPI Server.

The main purpose of this document is to illustrate new venues to build powerful logics to solve complex routing decisions with very flexible dial plans.

RestAPI Architecture

Representational state transfer (REST) is a style of software architecture. As described in a dissertation by Roy Fielding, REST is an "architectural style" that basically exploits the existing technology and protocols of the Web. RESTful is typically used to refer to web services implementing such an architecture.

Nowadays, REST is used for integration of totally autonomous systems or application. It is vendors preferred way to open their platform to third parties or even end user's creativity.

Today eBay, Salesforce, Amazon, Cisco, and many more consider REST API as a key component of their platforms.



The RestAPI loopback model with SIParator®



In our case, SIParator[®] will play both roles (RestAPI Client and RestAPI Server)

From now on we are going to call this architecture "Loopback RestAPI"

Deployment scenarios

In this section we will introduce the specifics of the use case we are going to use to illustrate our "Loopback RestAPI"

Use case description.

The case we are going to use for this illustration is a Multitenant Service Provider that is offering Hosted PBX services as well as PSTN brokerage.



For our example we are assuming there will be 3 customer's PBXs and 2 ITSP (SIP Trunk Providers) Each customer has one or more DID's associated, and can also own more than one PBX.

This table is a good way to present all possible combinations of PBX, DID's and ITSPs.

DID	Designated-PBX	ITSP
+19548668898	pbx1.edx-labs.com	itsp1edxlabs.pstn.twilio.com
+19548668002	pbx1.edx-labs.com	itsp1edxlabs.pstn.twilio.com
+19548668003	pbx2.edx-labs.com	itsp1edxlabs.pstn.twilio.com
+19548668004	pbx2.edx-labs.com	itsp1edxlabs.pstn.twilio.com
+19548668005	pbx3.edx-labs.com	itsp2edxlabs.pstn.twilio.com
+19548668006	pbx4.edx-labs.com	itsp2edxlabs.pstn.twilio.com
+19548668007	pbx4.edx-labs.com	itsp2edxlabs.pstn.twilio.com
+19548668008	pbx4.edx-labs.com	itsp2edxlabs.pstn.twilio.com

What this document will show later is how to implement a way to access this table in real-time during call setup from the dial plan.

We have to remember that there are other ways to deploy the routing for this case, but we want to illustrate how to implement with a more flexible approach. Having the routing rules based in a table such the one presented here, allows us to easily implement changes, additions or updates by just updating the table accordingly.

Proof of Concept Topology

Our lab to proof concept this case is shown below:



Figure 1: Deployment Layout

SIParator[®] will be deployed in a Data Center (In our case AWS Cloud) and ITSPs as well as PBXs will be located somewhere in the Public Internet

Also SIParator[®] will be behind AWS firewall in a public Subnet (DMZ).

Customer's PBXs are reachable via FQDN, in our example, and based in our sample table:

- pbx1.edx-labs.com
- pbx2.edx-labs.com
- pbx3.edx-labs.com
- pbx4.edx-labs.com

Service Providers (ITSPs) will be reachable also via FQDN. In our example even we are using the same provider (Twilio in our case – <u>www.twilio.com</u>), we will have separated trunks for each ITSP in our table:

- itsp1edxlabs.pstn.twilio.com
- itsp2edxlabs.pstn.twilio.com

Configuring SIParator[®] SBC

Pre-requisites

For this use case, validation has been done running SIParator[®] release 6.4.1 and the minimum licensing needed must include:

- Number of sip trunk concurrent session. Also known as CCS and must be at least the maximum number of concurrent SIP sessions we want the solution to support.
- Using the "Loopback RestAPI" approach we will not use Trunk Groups as all the calls will be managed directly in the Dial Plan.
- As we are going to use HTTP Services, at least 1 ACL license is needed to enable the HTTP Feature. Additional ACL might be needed only is SIP over WebSockets (WS or WSS) is used with Registrar Method, which is not our case.

If you have any doubts or questions about the best options for licensing, feel free to send your questions to support@educronix.com

No other licenses are needed to this specific use case. When transcoding is needed, there are no license needed as Transcoding feature is a built-in functionality purely based on software.

Make sure you are using one of the SIParator[®] appliances according to your expected workload, or a VM properly dimensioned if you are using Software SIParator[®]

Before initiating the deployment make sure you have:

• A Public IP address to be used exclusively for your SBC. It can be assigned in your firewall and properly routed to the SIParator[®] DMZ ip address.

Configuring IP Network Interfaces

SBC Interfaces will be assigned IP addresses for

- Outside Interface. The one sitting in the DMZ and associated to the public IP address.
- For the purpose of this use case, where ITSPs and PBXs are remotely located entities, no additional interface is needed.

SBC, in our case, is connected to the WAN/Internet through a DMZ connection.

In our case all interfaces are dedicated ethernet ports.

Configuring Interface

First, we will assign names to known IP addresses and ranges easily used later in the configuration. By known addresses we mean SIP Proxy addresses and Media Addresses for Customer's PBXs as well as ITSPs.

Admi	nistration	Basic Configura	tion Netwo	ork St	HTTP ervices	SIP Services	SIP Traffic	SIP Trunks	Q-TURN	Virtual Private Networks	Quality of Service	Logging and Tools	About Log out	
Netv Co	vorks and nputers	Default Gateways	All Interfaces	VLAN	EthO	Interface Status	PPPoE	Tunnels	Topology					
Ne	etworks	and Cor	nputers											
	Non		Subo	roup			Lowe	r Limit		(Upper Lii for IP rang	mit ges)	Interface/\/I_AN	
	Indi	lie	ວແມ່ຍູ	Jroup		DNS or IP A	Name ddress	IP	Address	DNS Na or IP Add	ame tress	IP Addres	SS	_
4	Twilio		Twilio Me	edia	~ [-	v
			Twilio Sig	gnalin	g 🗸 [-	~
	Twilio N	Media	-		•	34.203.25	50.0	34.2	203.250.0	34.203.251	.255	34.203.251.2	255 Ethernet0 (eth0 untagged)	v
			-		•	54.172.60	0.0	54.1	172.60.0	54.172.61.2	255 !	54.172.61.25	55 Ethernet0 (eth0 untagged)	v
			-		•	54.244.51	L.O	54.2	244.51.0	54.244.51.2	255	54.244.51.25	55 Ethernet0 (eth0 untagged)	v
e	Twilio S	Signaling	-		•	54.172.60	0.0	54.1	172.60.0	54.172.60.3	3 !	54.172.60.3	Ethernet0 (eth0 untagged)	v
۱L			-		~	54.244.51	L.O	54.2	244.51.0	54.244.51.3	3	54.244.51.3	Ethernet0 (eth0 untagged)	v
							<u>.</u>							
	pbx all		pbx1		•][-	~
			pbx2		Y [-	~
			pbx3		.								-	~
			pbx4		Y][-	~
÷	pbx1		-		•			_					Ethernet0 (eth0 untagged)	~
÷	pbx2		-		~								Ethernet0 (eth0 untagged)	~
÷	pbx3		-		~								Ethernet0 (eth0 untagged)	~
÷	pbx4		-		~								Ethernet0 (eth0 untagged)	~

- For PBXs we created an entry name for each Customer PBX IP address and then we created a name (pbx all) to aggregate all PBXs.
- For Twilio (ITSPs) we created a range for SIP Signaling and other ranges for Media, and then aggregated all of them under "Twilio" name.
- You should adapt to the ITSPs you are connecting to as well as PBXs.

Looking at our topology:



In our case,

- DMZ Network: 10.1.0.0/24
- Default Gateway: 10.1.0.1
- Eth0 IP 10.1.0.23
- Public IP: 34.195.141.39
- Public FQDN: acc.edx-labs.com

Ľ	Directly Connec	cted Networ	ks <u>(Help)</u>								
l	Name	Address	DNS Name	IP Address	Netmask / Bits	Network Address	Broadcast Address	Interface or Tunnel	VLAN Id	VLAN Name	Delete Row
I	eth0	Static 🗸	10.1.0.23	10.1.0.23	24	10.1.0.0	10.1.0.255	Ethernet0 (eth0) 🗸			

Static route for the default gateway:

Static Routing (H	<u>elp)</u>						
	Routed Network			Router			
DNS Name or Network Address	Network Address	Netmask / Bits	Dynamic	DNS Name or IP Address	IP Address	Interface or Tunnel	Delete Rov
default	default		- 🗸	10.1.0.1	10.1.0.1	Ethernet0 (eth0) 🗸	

Other Network related configurations

Let's assign the DNS server address. In our case we are going to use Google DNS 8.8.8.8

in Gate 1 other administrator(s)	currently logg	Adva ged in:	anced C	all C	ont	rol LAE	3
Administration Basic Configurat	tion Netwo	TP vices Se	SIP SIP rvices Traffi	SIF C Trun	ks (Q-TURN Vii	rtual F Netwo
BasicAccessConfigurationControlR	ADI' SNMP	Dynamic DNS Update	Certificates	ACME	TLS	Advanced Settings	SIPa Ty
General	V	ersion of S	oftware SI	Parate	or/Fi	irewall	
Name of this SIP Advanced Call I Default domain:	or: Cl SI Da So	neck for new Parator/Firev ate of last suc oftware versio	versions of s vall: ccessful vers on in use:	Softwa	re eck:	○ Ye Not av 6.4.1	es . ● vailab
·	P	olicy For P	ing To the	SIPar	ator	r	
IP Policy (Help) Discard IP packet Reject IP packets Reject IP packets TCP Reset	is (Never reply Only reply Reply to pi 	y to ping to ping to th ng to all IP a	e same address	e inte ses	erface	
DNS Servers <u>(Help</u>	<u>2)</u>						
No. Dynamic	DNS Na or IP Add	ame dress	Address De	elete R	ow		
1 - •	8.8.8.8	8.8	.8.8)			
Add new rows 1	rows.						_

You can also assign a name to this SIParator. The name will displayed in your browser tags.

Let's also assign an NTP server and setup time for the SIParator[®]. We are assuming to be located in EST time zone.

Administration Basic Configuration	Network HT Serv	TP SIP ices Services	SIP Traffic Tr	SIP unks Q-	TURN	irtual Privat Networks	e Quality Servic	of Loggir e and To
Save/Load Show Configuration	User Administration	Upgrade Table	Date and Time	Restart	License Server	Change Language		
Change Time Zone Dumont/DUrville (Antarct Dushanbe (Asia) East (Brazil) East-Indiana (US) Easter (Pacific) Easter (Pacific) Easter (Canada) Eastern (US) Change Date and Tit	(Help) Ca) Active Chai	e time zone: Ea nge time zone (Help) C	astern (US	5)	d Time	With NTE	9 (Help)	
Date: 2022-10-24 Time: 13:58:04 Set date and time man	ually	S M	ynchronize JTP Serv Dynamic - V Add new ro	e time wi ers To DN: or IP time.nis	ith NTP: Use If N S Name Addres it.gov	Yes (IP A I32.1	→ No abled ddress	Delete Row
Save Undo Look up	all IP address	es again						

Configuring SIP in SIParator®

Now we will setup all signaling related configuration for SIP.

Setup SIP Ports

Now we will need to associate ports to be used for SIP (UPD/TCP and/or TLS)

Go under SIP Services \rightarrow Basic Settings

in Gate 1 other administrator(s) c	urrently logged in.	Advanced	d Call Cor	ntrol LAB	}
Administration Basic Configurati	on Network Service	P SIP Services	SIP SIP Traffic Trunks	Q-TURN Vir	rtual Private Qua Networks Se
BasicSignalingMeSettingsEncryptionEncryption	dia Media option Transcoding	Interoperability	Sessions and Media	Remote SIP Connectivity	VoIP Survival
SIP Module (Help)	_				
Enable SIP module Disable SIP module	2				
SIP Signaling Port	s <u>(Help)</u>				
Active Port	Transport	Intercept	Allow From/To	Comr	nent Delete Row
Yes 🗸 5060	UDP and TCP 🗸	Yes 🗸	-	•	
No 🗸 5061	TLS 🗸	Yes 🗸	-	•	
Add new rows 1	rows.				
SIP Media Port Ra	nge <u>(Help)</u>				
Ports: 58024 -	60999				
Public IP Address	for NATed SIPa	rator (<u>Help)</u>			
DNS Name or IP Address	IP Address				
34.195.141.39	4.195.141.39				

- Make sure SIP Module is enabled
- By default, SIP Signaling port 5060 for UDP and TCP is already enabled and "Allow from" enables access from any network. We can later restrict this for only sources we trust for UDP or TCP.
- Port 5061 for TLS is non active. We are not going to activate it for this use case.
- As our SIParator[®] is sitting in a DMZ, the public IP is NATed and we need to write down the public IP address as indicated.

Setup SIP Filtering

At this point we will enable SIP filtering to allow SIP traffic only from known sources

inGate		Advar	ncec	l Call	Cor	ntrol L	AB			
Administration Basic Configuration	Network Servi	rP SI ices Servi	P ices 1	SIP Traffic T	SIP runks	Q-TURN	Virtual Priv Network	vate Qua is Se	ality of ervice a	Logging Ind Tools
Local Methods <mark>Filtering</mark> Registrar	Authentication	Accounts	STIR	Call Control	Dial Plan	Routing	Accounting	Time Classes	IDS/IPS	Test Agent
Sender IP Filter Rules	<u>(Help)</u>		_							
No. From Netw	vork Ac	tion	Delet	e Row	ſ	Default	Policy Fo	r SIP R	equests	s
1 Twilio	Proces	ss all 🗸	Þ_		(Proce	ss all			
2 pbx all	Proces	ss all 🗸			Ĺ	- Local	only			
Add new rows 1 row	vs.		1			🥑 кејес	t all			

- Allow (Process all) SIP traffic from Twilio (ITSPs) as well as any of customer's PBXs.
- Reject anything else

Setup SIP Monitoring

At this point we also want to monitor other SIP destination/origination IP addresses. In our case we will be monitoring the ITSPs as well as the PBXs IPs. SIParator[®] will monitor those IP's by sending periodically SIP OPTIONS requests.

SIP Servers To I	Monitor <u>(</u>	<u>Help)</u>	
Server	Port	Transport	Delete Row
pbx1.edx-labs.c		- •	
pbx2.edx-labs.c		- •	
pbx3.edx-labs.c		- •	
pbx4.edx-labs.c		- •	
itsp1edxlabs.ps		- •	
itsp2edxlabs.ps		- •	

- Add all PBX IPs or FQDNs pbx1.edx-labs.com pbx2.edx-labs.com pbx3.edx-labs.com pbx4.edx-labs.com
- All all ITSPs itsp1edxlabs.pstn.twilio.com itsp2edxlabs.pstn.twilio.com

Once all destinations (PBXs and ITSPs) are setup to accept traffic from this SIParator[®], you will see "online" status for each one of them.

n©ate	Ad	vanced	Call Cor	ntrol LAB		
Administration Basic Configuration Netw	work HTTP Services	SIP Services Tr	SIP SIP Trunks	Q-TURN Virtual Pr Networ	ivate Quality of ks Service	Logging and Tools About
Methods Filtering Registrar Auth	entication Acco	unts STIR	Call Dial Control Plan	Routing Accounting	Time g Classes IDS/IP	Test Agent <mark>Status</mark>
Monitored SIP Servers			_			
Server	Port Transpo	t Status	1			
pbx2.edx-labs.com	UDP	Online				
pbx1.edx-labs.com	UDP	Online				
itsp2edxlabs.pstn.twilio.com	UDP	Online				
pbx3.edx-labs.com	UDP	Online				
itsp1edxlabs.pstn.twilio.com	UDP	Online				
pbx4.edx-labs.com	UDP	Online	1			
-						

Setup Call Control for RestAPI

In this case we want to enable the Ingate to send Restfull API requests to a Web Service. Also, in our case the Web Service server happens to be the same SIParator[®] (Itself)

We have enabled an FQDN that also resolves on the public IP associated to this SIParator®

(acc.edx-labs.com)

It is clear then that this FQDN points to the SIParator® itself.

In order to avoid conflicts to other services hosted by the SIParator[®] in port tcp 80 (i.e. ACME), we will use port 8080 instead for RestAPI Serevr functionality.

inG	ate			Advar	nceo	d Call	Coi	ntrol L	AB							
Administr	ation Cor	Basic Ifiguration	Network HT Serv	TP SI fices Servi	P ices	SIP Traffic	SIP Trunks	Q-TURN	Virtual Priv Network	vate Qu s Si	ality of ervice a	Logging and Tool	About	Log out		
Methods	Filtering	Local Registrar	Authentication	Accounts	STIR	Call Control	Dial Plan	Routing	Accounting	Time Classes	IDS/IPS	Test Agent	Status			
REST	API Sei	vers <u>(H</u>	elp)													
	ID			Prefix						Suff	ix			Request Timeout	Cache Lifetime	Delete Row
1		http://	acc.edx-labs.c	om:8080/	trunki	ng.json								60	0	
Add n	ew rows	1 rov	/S.				<u> </u>									_

- We have selected ID=1 (to be able to refer to \$curl1 function from the dial plan. It will always replace http://acc.edx-labs.com:8080/trunking.json as a prefix to any expression in the forward-to in the dial plan.
- Leave all other fields with default preloaded values.

For debugging purposes enable verbose logging.



At this point we have configured the RestAPI Client Role

Configure HTTP Services to provide RestAPI Server behavior

You should have already installed at least 1 ACL License to be able to ebale HTTP Services in the Management interface

You can check so under the About tag:

Licenses
 10 SIP Registrar Users 10 Concurrent Calls SIP Trunk Sessions (of max unlimited) 10 Remote User SIP Sessions (of max unlimited) 1 Advanced Client Licenses 1 Trunk Group Trunk Group 1 can have max 10 Concurrent Calls SIP Trunk Sessions 0 Q-TURN Sessions

Enable HTTP Services and all a new row in the "Local Files" Section:

inGo	ate	Advanced Call	Control LAB		
Administr	ation Basic Configuration	letwork HTTP SIP SIP Services Services Traffic	SIP Trunks Q-TURN Virtual Private Quali Networks Serv	ty of Logging vice and Tools About Log of	ut
• HT • Th	TP Storage is enable is page contains add	ed, but no Repositories are defined. itional errors.			
Storage a Tunnels	webSockets and HTTP				
Stora	de Repositories a	nd Tunnels (<u>Help)</u>			
● En: ○ Dis	able				
Loca	al Files (Utilization	: 0.00% Space available: 2.0	MiB) (<u>Help)</u>		
A file	hosted locally on the	unit. Upload/Edit the file contents.	Add a file to a File Group below.		
	Name	Path	File Name	File	Information Re
No f	ile exists.				
Tru	nking	1	trunking.json	Upload Download E	dit file

- Enable Storage Repositories and tunnels
- Add a new raw and assign a name (It could be any name with no blank spaces)
- Select the root folder (/)
- Assign a name to the file as it will be named inside the folder (trunking.json) which matches with the name used in the Call Control prefix for curl1.
- We name it trunking.json as we are going to use a json script content

To create the content and load it in the file we will take advantage of our excel file shown before.

There are many ways you can enter or create a json file, but iin our case we are using Excel and also a public add-in you can install in your Excel application.

To obtan and install this add-in you can go here: <u>Get Started — Excel-to-JSON 1.4.0.0 documentation</u> (wtsolutions.cn)

Once it is installed, you can convert your table to json script.

File Hom Launch Excel- to-JSON Excel-to-JSON	e Insert Page Layout Formulas Dat	a Review View Automate Developer	Help QuickBooks <u>Excel-to-JSON</u>		다 Comments 전 와 Share
F8 -	\cdot : $\times \checkmark f_x$				
A	ВСС	D	F G H	I J 🔶 Excel to J	SON -
1 Nr 2 3 4 5 6 7 8 9 10 11 12 13	DD Designated-PBX 1 +19548668002 pbx1.edx-labs.com 1 +19548668002 pbx1.edx-labs.com 2 +19548668003 pbx2.edx-labs.com 3 +19548668005 pbx3.edx-labs.com 4 +19548668005 pbx4.edx-labs.com 4 +19548668006 pbx4.edx-labs.com 4 +19548668007 pbx4.edx-labs.com 4 +19548668008 pbx4.edx-labs.com 4 +19548668008 pbx4.edx-labs.com	ITSP itsp1edxlabs.pstn.twilio.com itsp1edxlabs.pstn.twilio.com itsp1edxlabs.pstn.twilio.com itsp2edxlabs.pstn.twilio.com itsp2edxlabs.pstn.twilio.com itsp2edxlabs.pstn.twilio.com itsp2edxlabs.pstn.twilio.com		An Excel A Examples i If you have conversion Your Donal JSCN-to-E:	Add in that Convert Excel to JSON. add in that Convert Excel to JSON. and Documentations: a BIG datasheet, it is recommended that in batches. tion is important to maintain the addin ser axcel Excel Add-in available here
13 14 15 16 17 18 19				Your Dor server liv Your Fee	Go nation is important to maintain the e. dback is important to improve th
(→ Cus Deads FR 10	stomers Table Sheet1 Cutomer ID PBX Custon	mer ID DID PBX DID 🕀	:	R Disolary Settions	► = + 12

- Select Excel-to-JSON tag
- Click on Launch button
- Select the table
- Click on "Go" button

Scroll down the Excel to Json screen in the left side of the spreadsheet. You'll see the jscon script generated and also a " Copy to Clipboard" button.

In your recenty=ly add file in the SIParator, click on the edit button, and paste there the content to created with the Excel Add-in.

Local Files (Utilization	: 0.00% Space available: 2.0	MiB) <u>(Help)</u>			
A file hosted locally on the	unit. Upload/Edit the file contents.	Add a file to a File Group below.			
Name	Path	File Name	File	Information)e R
No file exists.					
Trunking	1	trunking.json	Upload Download Edit	No current ile	
Add new rows 1 rows	S.				

Edit the local file "Trunking" below. [{"DID": "+19548668898", "Designated-PBX": "pbx1.edx- labs.com", "ITSP": "itsp1edxlabs.pstn.twilio.com"}, {"DID": "+19548668002", "Designated-PBX": "pbx1.edx- labs.com", "ITSP": "itsp1edxlabs.pstn.twilio.com"}, {"DID": "+19548668003", "Designated-PBX": "pbx2.edx- labs.com", "ITSP": "itsp1edxlabs.pstn.twilio.com"}, {"DID": "+19548668004", "Designated-PBX": "pbx2.edx- labs.com", "ITSP": "itsp1edxlabs.pstn.twilio.com"}, {"DID": "+19548668005", "Designated-PBX": "pbx2.edx- labs.com", "ITSP": "itsp1edxlabs.pstn.twilio.com"}, {"DID": "+19548668005", "Designated-PBX": "pbx3.edx- labs.com", "ITSP": "itsp2edxlabs.pstn.twilio.com"}, {"DID": "+1954866805", "Designated-PBX": "pbx3.edx- labs.com", "ITSP": "itsp2edxlabs.pstn.twilio.com"}, {"DID": "+195486805", "Designated-PBX": "pbx3.edx- labs.com", "ITSP": "Itsp2edxlabs.pstn	
<pre>[{"DID":"+19548668898","Designated-PBX":"pbx1.edx- labs.com","ITSP":"itsp1edxlabs.pstn.twilio.com"}, {"DID":"+19548668002","Designated-PBX":"pbx1.edx- labs.com","ITSP":"itsp1edxlabs.pstn.twilio.com"}, {"DID":"+19548668003","Designated-PBX":"pbx2.edx- labs.com","ITSP":"itsp1edxlabs.pstn.twilio.com"}, {"DID":"+19548668004","Designated-PBX":"pbx2.edx- labs.com","ITSP":"itsp1edxlabs.pstn.twilio.com"}, {"DID":"+19548668005","Designated-PBX":"pbx2.edx- labs.com","ITSP":"itsp1edxlabs.pstn.twilio.com"}, {"DID":"+19548668005","Designated-PBX":"pbx3.edx- labs.com","ITSP":"itsp2edxlabs.pstn.twilio.com"},</pre>	
<pre>{"DID":"+19548668006","Designated-PBX":"pbx4.edx- labs.com","ITSP":"itsp2edxlabs.pstn.twilio.com"}, {"DID":"+19548668000","Designated-PBX":"pbx4.edx- labs.com","ITSP":"itsp2edxlabs.pstn.twilio.com"}, {"DID":"+19548668008","Designated-PBX":"pbx4.edx- labs.com","ITSP":"itsp2edxlabs.pstn.twilio.com"}]</pre>	
Abort	ave

- Paste the content in the edit screen.
- Save the file

The file is now created

Local Files (Utilization	n: 0.04% Space available: 2.0	MiB) (<u>Help)</u>					
A file hosted locally on the	e unit. Upload/Edit the file contents.	Add a file to a File Group below.					
Name	Path	File Name		File		Information	Delete Row
Trunking	1	trunking.json	Upload	Download	Edit	Size: 753 bytes MIME type: text/plain SHA256: eca738a5811971e70d5a38880dfddbcd8a86bb66166b000f273c63c67645aa77	
Add now rows 1 row	NC			-			_

If you prefer you can use any other method to generate the JSON content. Here you can also use XML if it is of your preference. SIParator supports both (JSON or XML content) for call control purposes.

Next step is to create a local file group that contains the recently created file.

Local File Groups (Help)	1		
A group of files that are hoste	ed locally. A File	e is defined a	bove. Attach a file group to a Repository that is defined below.
Name	File	Delete Row	
+ CallControl	Trunking 🗸		
Add new rows 1 groups	with 1 rows	s per group.	

- Assign a name.
- Pull down and select he file we just created

Now, we need to create the local endpoint to enable access to this repository via port 8080 as we decided before.

Before doing that we will create a network name associated to the public IP of the SIParator[®] to restrict this RestAPI request only coming from itself.

letworks and Defaul Computers Gatewa	t All ys Interfaces	VLAN E	Interfa th0 Statu	ice Is PPPoE	Tunnels	Topology				
Networks and C	omputers									
h	Curls			Low	er Limit		Upper (for IP ra	Limit anges)	Interface D.C. Alt	
Name	Sub	group	DN or II	NS <mark>Name</mark> P Addres	s IP /	Address	DNS Name or IP Address	IP Address	Interface/vLAN	
+ acc]-		✔ 34.195	5. <mark>1</mark> 41.39	34.1	95. <mark>141</mark> .39				``
	-181					_	12		5e	

Now let's add the local endpoint

	Local Endpoints (Hel	<u>p)</u>								
L	A local endpoint serves a	s an entry po	int for locally and rem	otely hosted	l files. It can also	serve as an entry	point for HTTP tunnels.	A Repository must ha	ave a local end	point defi
L	Name	Protocol		Port	Server	Peer Verification	TLS	Allow From	Delete Row	
	Name	FIOLOCOI	IF Address	FOIL	Certificate	Trusted CAs	Settings	Allow From	Delete Row	
L	CallControlEP	HTTP 🗸	eth0 (10.1.0.23) 🗸	8080	- •	- 🗸	- ~	acc 🗸		
L	Add new rows 1 row	NS.								

- Add a row
- Assign a name (i.e. CallControlEP
- Select the protocol (HTTP in our case)
- Select the network interface where the RestAPI requests will be served
- Restrict access only from acc network recently created

Finally let's create the repository hosting the file and how it will be accessed.

Go to Repositories and Tunnels section.

Repositories and Tur	nnels <u>(Help)</u>						
A repository defines stor	age for local and/or r	remote files. Def	ine Local/R	emote Endpoi	nts and Local File G	roups above. HT	TP tunnels via th
Local		Local	Local Remote		Tunn	Delete Dow	
Name	Endpoint	File Group	Endpoint	Methods	Allow To	Ports	Delete Row
CallControlREPO	CallControlEP 🗸	CallControl 🗸	- 🗸	DEFAULT 🗸	- *		

- Add a new row
- Assign a name (i.e. CallControlEP)
- Select the Local Endpoint to use
- Select the File Group to expose

At this point we have configured the RestAPI Server Role.

We are ready now to move to de Dial Plan configuration and setup

Configure Dial Plan

Using Dial Plan we will be able to route inbound and outbound traffic. It includes traffic from any ITSP and properly route to the associated Customer's PBX based on number dialed from the originator (DID in the Request-URI) as well as traffic sent from any Customer's PBX to the appropriate ITSP based on the number contained in the P-Asserted-Identity in the outbound call.

First you'll need to enable Dial Plan.



Then we need to create Matching rules for From header and Request URI. This will help on building routing rules.

Matching Fror	n Header <u>(Help)</u>					
Namo	Use T	'his	Or This	Transport	Notwork	Delete Deu
Name	Username	Domain	Reg Expr	Transport	Network	Delete Row
From ITSPs	*	*		Any 🗸	Twilio 🗸] 🗅
From PBXs	*	*		Any 🗸	pbx all 🗸] 🗅

• Add one row to match SIP traffic coming from the ITSP

- Assign a name to each one
- We'll use wildcards ("*") for Username and Domain
- You can select specific transport protocol, but in our case we'll keep it open to "Any".
- Restrict to the Network names where the traffic could be coming from

Now, let's complete Request-URI matching

Matching Requ	iest-URI <u>(Help)</u>					
Name			Use This			Or This
Name	Prefix	Head	Tail	Min. Tail	Domain	Reg Expr
To SIParator			- •			sip:(.*)@(34.195.141.39 i

- Add one raw and assign a name
- Select "-" for Tail
- Use regular expressions and match either with the Public IP of the SIParator[®] or the FQDN.
 sip:(.*)@(34.195.141.39|acc.edx-labs.com)

Now we will create the "Forward to" destinations, one to route traffic to Customer PBX, and the second one to the appropriate ITSP based on the phone number dialed (PSTN \rightarrow PBX) or used as caller ID (PBX \rightarrow PSTN)

orward To <u>(Help)</u>									
Namo	No	Use This	Or	This		Or This	Or This	Lico Aliao ID	Delete Dow
Name	NO.	Account	Replacement Domain	Port	Transport	Reg Expr	Trunk	USE Allas IP	Delete Row
To Customer Pt	1	- 🗸			- •	sip:\$r1@\$curl1(- •	- 🗸	
+ To PSTN	1	- 🗸			- •	sip:\$r1@\$curl1(- •	- 🗸	

- Add two rows
- Assign names to each one
- Use the following regular expression for PSTN \rightarrow PBX

sip:\$r1@\$curl1(_XPATH//*[DID="\$r1"]/Designated-PBX/text());b2buawm;transport=udp

• Use the following regular expression for PBX \rightarrow PSTN

sip:\$r1@\$curl1(_XPATH//*[DID="\$(P-Asserted-Identity.user)"]/ITSP/text());b2buawm;transport=udp

Lets understand in detail what we are doing with the regular expressions we just introduced.

First, we are assuming:

- 1) ITSP if fully compliant with E164 for any phone number used in any SIP URI (i.e. Request-URI, From, To, Contact and P-Asserted-Identity headers)
- 2) The PBX sends the Caller ID in the "P-Asserted-Identity" header

3) Reference to \$Curl1 are using The Call Control definitions we created before

sip:\$r1@<mark>\$curl1(_XPATH</mark>//*[DID="\$r1"]/Designated-PBX/text());b2buawm;transport=udp

- We are using XPATH navigation/parsing function supported by SIParator[®], which will help navigate in our JSON file hosted by HTTP Services.
 - \circ //* \rightarrow allows us to search all levels in the file starting in the root position.
 - [DID="\$r1"] → look for the row that matches DID field with the value obtained from the variable \$r1. \$r1 is obtained in the first parenthesis match in the Request-URI matching rule
 - O /Designated-PBX/text() → describes which field will be used when matching DID field. In our case the field used is "Designated-PBX". Then "/text()" means the result will be converted to plain text.

So, the XPATH here will look in the JSON file for the value of Designated-PBX field where the DID matches the number that was called and matched in the Request-URI.

INVITE sig +19548668898 Dacc.edx-labs.com SIP/2.0 Record-Route: <sip:54.244.51 1:Ir> From: <sip:+12404018146@itsp1 vilio.com:5060:isup-oli=62:pstr To: <sip:+19548668898@acc.edx-lal CSeq: 537556 INVITE Max-Forwards: 64 P-Asserted-Identity: <sip:+12404018146@206.147.88.39:5060> Diversion: <sip:+19548668898@twilio.com>;reason=unconditional Call-ID: 69c67dd05e0f8a25d6b20af8fa0df4af@0.0.0.0 Via: SIP/2.0/UDP 54.244.51.1:5060;branch=z9hG4bKfd89.7eea54c8637e0bfe Via: SIP/2.0/UDP 172.18.64.54:5060;rport=5060;branch=z9hG4bKf1c3cab5-al Contact: <sip:+12404018146@172.18.64.54:5060;transport=udp> Allow: INVITE.ACK.CANCEL.BYE.REFER.NOTIFY.OPTIONS

 \$curl1(...) will use the Call control to look for the web service and return the value obtained in the previous point, so the result will be = *pbx1.edx-labs.com*

	- 1	DID		Designated-PBX	ITSP	
		+19	548668898	pbx1.edx-labs.com	itsp	1edxlabs.pstn.twilio.com
		+19	545668002	pbx1.edx-labs.com	'tcn'	1edxlabs.pstn.twilio.com
			548668003	pbx2.edx-labs.com	itsp	Obtained PBX om
	Matched	DID	548668004	pbx2.edx-labs.com	itsp	address om
1	_	+19	548668005	pbx3.edx-labs.com	itsp2	2edxlabs.pstn.twilio.com
		+19	548668006	pbx4.edx-labs.com	itsp2	2edxlabs.pstn.twilio.com
		+19	548668007	pbx4.edx-labs.com	itsp	2edxlabs.pstn.twilio.com
		+19	548668008	pbx4.edx-labs.com	itsp	2edxlabs.pstn.twilio.com

After replacing curl1 result the expression will look like:

sip:\$r1@pbx1.edx-labs.com;b2buawm,transport=udp

• Then \$r1 will be replaced again for the dialed number +19548668898, and the final destination to route the call will be:

sip:+19548668898@pbx1.edx-labs.com;b2bua;transport=udp

INVITE sip:+19548668898@pbx1.edx-labs.com;transport=udp SIP/2.0
Via: SIP/2.0/TCP 34.195.141.39:5060;branch=z9hG4bK4b5f4a8d73cb93a82af24
Session-Expires: 14400
Via: SIP/2.0/UDP 34.195.141.39:5060;branch=z9hG4bKb247ca5fd041947436f2
From: <sip:+12404018146@itsp1edxlabs.pstn.twilio.com:5060;isup-oli=62;pstn-j< td=""></sip:+12404018146@itsp1edxlabs.pstn.twilio.com:5060;isup-oli=62;pstn-j<>
To: <sip:+19548668898@pbx1.edx-labs.com;transport=udp></sip:+19548668898@pbx1.edx-labs.com;transport=udp>
Call-ID: 13ca1134e292f71225b02515de8e3714@sipgt-57d6798b
CSeq: 537556 INVITE
User-Agent: SIParator/6.4.1
Contact: <sip:eiuldvkckgabnysionzy6ypb1ieeligcwpjby-iir684.@34.195.141< td=""></sip:eiuldvkckgabnysionzy6ypb1ieeligcwpjby-iir684.@34.195.141<>
Supported: timer, replaces, path, histinfo, 100rel

sip:\$r1@\$curl1(_XPATH//*[DID="\$(P-Asserted-Identity.user)"]/ITSP/text());b2buawm;transport=udp

In this case, instead of looking for the \$r1 (dialed number) will look for \$(P-Asserted-Identity.user) value in the call coming from the PBX. It will be matched against DID field in the table.

INVITE sip:+12404018146@acc.edx-labs.com SIP/2.0 Via: SIP/2.0/UDP 34.233.24.99:5060:branch=z9hG4bKbc19e4fb18/
Session-Expires: 14400
Via: SIP/2.0/UDP 34.233.24.99:5060;alias;branch=z9hG4bK1e5b45
From: "+19548668898", raisy 105490090909 and adv laboration da
To: <sip:+1240401814< th=""></sip:+1240401814<>
Call-ID: 7d69809f626c
CSeq: 102 INVITE
User-Agent: Labs/AWS
P-Asserted-Identity: sip:+19548668898@127.0.0.1
Contact: <sin:trunk2.1.1@34.233.24.99></sin:trunk2.1.1@34.233.24.99>
Supported: timer, replaces, path, histinfo, 100rel
AN ACK CANCEL BUE THESE NAMES NOTICE COTIONS OF

And we will extract the value of the "ITSP" field in text format

DID	Designate	ed-PBX	ITS	SP		
+19548668898	pbx1.edx-	labs.com	its	p1edxlabs.p	stn.twilio	.com
+19548668002	pbx1.edx-	labs.com	its	p1edxlabs.	stn.twilio	.com
+195486680	latched PAI	labs.com	its			om
+195486680 <mark>c</mark> .	PRALICUA	labs.com	its	Returne	d ITSP	com
+19548668005	pbx3.edx-	labs.com	its	addro	ess	com
+19548668006	pbx4.edx-	labs.com	its	p2edxlabs.p	stn.twilio	.com
+19548668007	pbx4.edx-	labs.com	its	p2edxlabs.p	stn.twilio	.com
+19548668008	pbx4.edx-	labs.com	its	p2edxlabs.p	stn.twilio	com

For useful information about regular expressions and XPATH you can visit the following links:

XPath Tutorial (w3schools.com)

How To use Generic Header Manipulation.pdf (ingate.com)

regex101: build, test, and debug regex

Ingate Reference Guide (call Control)

Ingate Reference Guide (HTTP Services)

Next step is to complete the actual dial plan.

Dial plan execution happens in the Dial Plan Table

ß	Dial Plar	n <u>(Help)</u>											
L	No	From Header	Request-URI	Action		Forward To		Add F	Prefix	ENUM Root	Time	Comment	De
		Trontricader	ricquest ora	Action		T OTHAI UTO	F	orward	ENUM	Litomittoot	Class	Connient	R
L	1	From ITSPs 🗸	To SIParator 🗸	Forward	~	To Customer PBX 🗸				- 🗸	- 🗸		C
L	2	From PBXs 🗸	To SIParator 🗸	Forward	~	To PSTN 🗸				. 🗸	- 🗸		C
	-						_						

- Add two rows to the dial plan table
- The first will match From Header based on the matching rule named "From ITSPs" and match Request-URI named "To SIParator". When that happens, call will be routed using "Forward to" named "To Customer PBX"
- The second will match From Header based on the matching rule named "From PBXs" and match Request-URI named "To SIParator". When that happens, call will be routed using "Forward to" named "To PSTN"

How easy is to maintain this Configuration?

Semi-manual update

At this point, the configuration we just created is very easy to maintain. Maintenance means:

- Change attributes assigned to a given DID (i.e. change ITSP hosting it, will be associate to another PBX, etc...)
- Add new or delete DIDs.
- Add new or delete customers
- Add new or delete ITSPs
- Etc...

This is the first beauty of this approach ("easy to maintain")

First, any change, add or remove rows can be done just in the table:

- Update your Excel file
- Execute the add-in "Excel-to-JSON".
- Copy and paste in the HTTP Services local file.



If any of the updates done added a new ITSP address or PBX address, make the appropriate adjustments in the Network \rightarrow Networks and Computers Section.

works and Default Imputers Gateways etworks and Co	All Interfaces VLAN Ethl mputers	Interface Status PPPoE Tu	innels Topology			
News	Culture	Lower	.imit	Upper I (for IP ra		
Name	Subgroup	DNS Name or IP Address	IP Address	DNS Name or IP Address	IP Address	Interface/vLAN
Twilio	Twilio Media 🗸					-
	Twilio Signaling 🗸					
Twilio Media	· •					Ethernet0 (eth0 untage
	- 🗸					Ethernet0 (eth0 untage
	- 🗸					Ethernet0 (eth0 untagg
 Twilio Signaling 	- 🗸					Ethernet0 (eth0 untage
	- 🗸					Ethernet0 (eth0 untage
acc	- 🗸					-
+ pbx all	pbx1 🗸					-
	pbx2 🗸					-
	pbx3 🗸	_				-
	pbx4 🗸	_				-
+ pbx1	- •					Ethernet0 (eth0 untage
+ pbx2	- •					Ethernet0 (eth0 untage
b) pbx3	· •					Ethernet0 (eth0 untage

If new endpoints were added such as new PBX, or new ITSP proxy addresses, you should add them to the SIP Services \rightarrow Basic Settings, under the Servers to Monitor table.

Server	Port	Transport	Delete Row
pbx1.edx-labs.(- 🗸	
pbx2.edx-labs.(- 🗸	
pbx3.edx-labs.(- 🗸	
pbx4.edx-labs.(- 🖌	
itsp1edxlabs.ps		- 🖌	
itsp2edxlabs.ps		- 🗸	

Automated Updates option

In case you want to create a more automated way to update, you can do 2 things:

Instead of using a manual process to maintain an Excel table and then manual cut and paste in the HTTP Services local file section, you can implement RestAPI provisioning and use your own scripting of implement Ingate's SDK

For more details see here:

- <u>https://account.ingate.com/manuals/6 4 1/reference guide 6 4 1.html# access control</u> (look on how to enable RestAPI clients)
- https://account.ingate.com/manuals/6_4_1/reference_guide_6_4_1.html#_python_sdk
- <u>https://account.ingate.com/manuals/6_4_1/reference_guide_6_4_1.html#_command_line_ref</u> erence

This way with some development you can automate updates to your base of customers, ITSPs etc... with a totally external application.

Disclaimers

SIParator[®] and Ingate[®] are Trademarks of Ingate System AB

This documentation is intellectual property of Educronix LLC and is copyright protected

Help and Support