SANGOMA

Netborder SS7 to VoIP Gateway

User Manual

Date: Aug 22 2012
Version: 1.03

Revision History

Document Revision	Date	Description of Changes
1.03	Aug 22 2012	Pinout label
1.02	Aug 22 2012	VLAN, Factory Reset, Static Routes, Eth Options, usb console, DC PSU info
1.01	Aug 19 2012	Added extra diagrams, Media, SIP, Relay, Dialplan, Update, Cables, Appendix
1.00	Aug 2012	Initial revision of the document.

Conventions
This font indicates screen menus and parameters.
<> indicates keyboard keys (<Enter>, <q>, <s>).

NOTE
Notes inform the user of additional but essential information or features.

CAUTION
Cautions inform the user of potential damage, malfunction, or disruption to equipment, software, or environment.

Sangoma Technologies provides technical support for this product.
Tech-support e-mail: techdesk@sangoma.com

This page is intentionally blank.

Sangoma

Netborder SS7 to VoIP GW User Manual

Contents

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

1. Product Overview

The NetBorder SS7 to VoIP Gateway is Sangoma's Carrier Class TDM to SIP VoIP Gateway product. For short, it is often referred to as NSG.

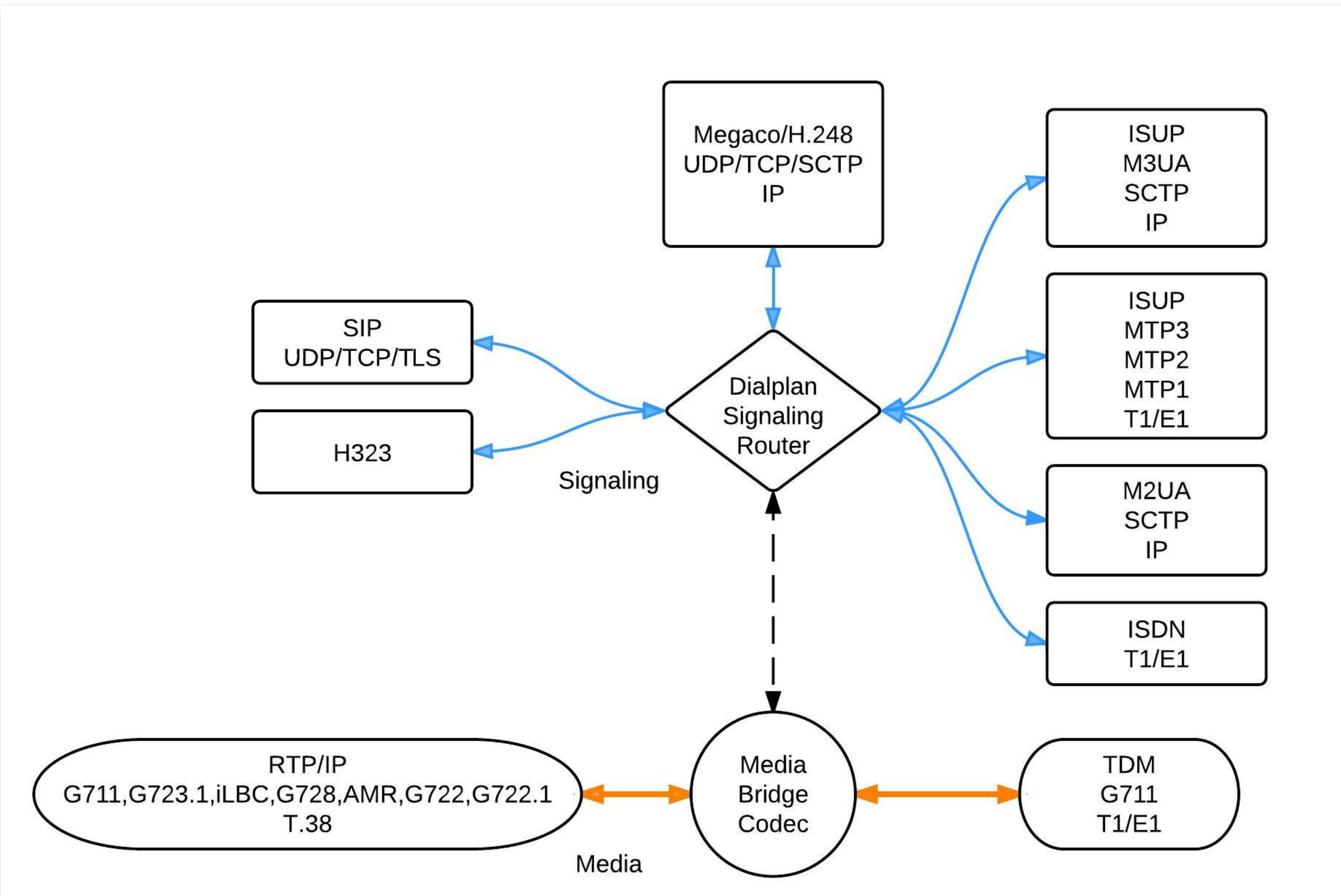
Features / Advantages

- Any to any switching gateway.
  - Ability to run all endpoints/protocols at the same time on single appliance
  - SS7, Sigtran, SIP, H323, Megaco Media Gateway, Signaling Gateway
  - Flexible dial plan to route from any endpoint to any endpoint
- Scalable and very high density
  - Up to 32 E1 per server
  - Can scale up to 288 E1s in relay mode where multiple systems act as one
  - Transcoding available on all channels
- Extensive VoIP Signaling
  - SIP, H323, Megaco/H248
- Full featured SS7/Sigtran Signaling
  - SS7 ISUP Signaling with several national variants
  - ITU, ANSI, Bellcore, France, UK, China, India and Russia
  - Sigtran, M3UA, M2UA
  - Sigtran signaling gateway
- ISDN signaling
  - Q931,QSIG,
- Faxing and Media Support
  - Pass-through
  - T.38
- Wide range of narrowband and wideband codecs supported
  For any-to-any codec transcoding
  - G711,G729,AMR

- Robust implementation with distribution, failover and redundancy

Any to Any Signaling and Media Gateway

- Route any signaling traffic from eny signaling endpoint simultaneously.
- Ability to run all protocols together at the same time.
- Route media with transcoding/dtmf/T.38 to/from end media endpoint.



TDM T1/E1 Interfaces

- Minimum 4 T1/E1
- Maximum 32 T1/E1 (960 ports) per appliance
- Transcoding supported on all channels
- Extend capacity over 960 ports via ISUP relay feature and multiple appliances.

Ethernet Network Interfaces

- 2 Gigabit network interfaces

VoIP Protocols

SIP

- SIP V2 / RFC3261 RFC 3261 Session Initiate Protocol
- RFC 2976 SIP INFO Method
- RFC 3398 ISUP-SIP Mapping
- RFC 3515 Refer Method
- RFC 2327 Session Description Protocol
- RFC 3581 An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing
- RFC 3892 Referred-By Mechanism
- RFC 3891 "Replaces" Header
- RFC 3551: RTP/AVP
- RFC 3515: REFER
- RFC 2617: HTTP Digest Authentication
- SDP Bypass
- NSG exports all SS7 parameters via SIP custom X headers.

Megaco/H248 & MGCP

- Media Gateway Control Protocol Version 1.0, Internet RFC3435
- PacketCable Network-Based Call Signaling (NCS) Protocol Specification
- PacketCable PSTN Gateway Call Signaling (TGCP) Protoc ol Specification
- MGCP Basic Packages – RFC 3660
- MGCP Fax Package – draft-andreasen-mgcp-fax-00.txt

H323

Call Handling

- H.225.0 : Call signaling protocols and media stream packetization for packet-based multimedia communication systems
- H.245 : Control protocol for multimedia communication
- H.235, H.450, H.460

DTMF support

- RFC 2833/4733 - "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals"
- In-band DTMF detection/generation

TDM Protocols

SS7

- ISUP, MTP3, MTP2, MTP1, M3UA, M2UA, Relay
- Variants
  - ITU, ANSI, Bellcore, UK, China, France Spirou, India and Russian
- MTP2
  - ITU 88 & 92, ANSI 88 & 92, Peoples Republic of China
- MTP3
  - ITU 88 & 92 & ETSI, ANSI 88 & 92, 96 & Telcordia (including ANSI MTP3-B), China
- ISUP
  - ITU 88, 92 & 97, 2000, Telcordia 97, ANSI 88, 92, 95 and ETSI v2,v3
  - SPIROU, China, UK, Russia, India

ISDN

• CCITT 88,	User & Network Side PRI/BRI
• AT&T 4ESS	User Side - PRI, Network Side - PRI
• 5ESS	User Side - PRI/BRI, Network Side - PRI/BRI
• DMS-100	User & Network Side - PRI/BRI
• ETSI	User & Network Side - PRI/BRI
• Australian Telecom	User Side - PRI/BRI and Network Side - PRI
• National ISDN-1	User Side - BRI
• NTT	User & Network Side - PRI/BRI
• National ISDN-2	User & Network Side - PRI
• Q.SIG	(PRI)
• LAPD & TEI Management	

Call Routing

Configurable and extendable XML-based dial plan and routing rules XML Dialplan can be used to create complex routing scenarios between SIP and TDM.

- Call routing based on any call parameter present in a SIP or SS7 IAM message.
- Deep integration with signaling stacks
- Ability to use external applications to build complex routing logic*

Media Processing & Transcoding

Wide range of codecs supported for any to any codec negotiation.

- G.711
- G.723.1
- G.726
- iLBC
- G.729AB
- GSM
- G.722
- AMR
- G.722.1

Echo Cancellation & VQE

Telco grade hardware based echo canceling and Voice processing

- G.168-2002 with 128ms tail
- Noise cancellation
- DTMF Removal
- DTMF Detection
- FAX Detection
- Automatic Gain Control

DTMF Detection and Generation

Sangoma NSG gateway supports multiple DTMF internetworking scenarios.

- RFC2833 Tone Relay
- In-band
- SIP INFO
- Hardware and software DTMF detection and generation

Management and Configuration

Sangoma NSG configuration, operation and troubleshooting is designed to be flexible.

- Web GUI
- Command line interface
- Call detail records in XML format
- Detailed logs with user configurable file size and auto rotation

Monitoring

- SNMP v1,2,3
- RTCP

Accounting

- Radius

Shipping Options

SKU	DESCRIPTION
SS7-NSG-AP04	Up to 4 E1/T1, ISUP to SIP, codec support, 4 signaling links, up to 12 point codes
SS7-NSG-AP08	Up to 8 E1/T1, ISUP to SIP, codec support, 8 signaling links, up to 12 point codes
SS7-NSG-AP16	Up to 16 E1/T1, ISUP to SIP, codec support, 16 signaling links, up to 12 point codes
SS7-NSG-AP32	Up to 32 E1/T1, ISUP to SIP, codec support, 32 signaling links, up to 12 point codes

Support and Professional Services

Sangoma Engineers are here to support your success. Whether you need technical support and software maintenance, training, consultation and installation services, Sangoma can help you. Please contact your Sales representative for more information.

2. Product Options

NetBorder SS7 to VoIP Gateway Appliance

Fully integrated Industrial grade telco appliance running a customized OS, Netborder SS7 to VoIP application and TDM interfaces configured and installed by Sangoma.
NSG Appliance provides a full-featured, carrier-class VoIP deployment while leveraging the flexibility and cost effectiveness of standard computing platforms.

Hardware Specifications

- Industrial grade telecom appliance
- Size: 1U and 2U - 19'' Rackmount
- Min Capacity: 4 T1/E1   (1U)
- Max Capacity: 32 T1/E1 (2U)
- Power: AC, DC, Redundant
- AC Power Supply (Single)
    -
- DC Power Supply (Redundant)
    - The Input Current for -48VDC, is 12.0A (RMS).
    - With Inrush Current of 20.0A MAX.
- Depth: 20''
- Weight: 36lb
- Full Spec on Sangoma Site

3. Getting Started

Look into the Box

The first three tasks for installing and operating the Netborder SS7 to VOIP Gateway are

- Unpack
- Inspect
- Power up.

Carefully inspect the NSG Appliance for any damage that might have occurred in shipment.

If damage is suspected, file a claim immediately with the carrier, keep the original packaging for damage verification and/or returning the unit, and contact Sangoma Customer Service.

What is included in the box

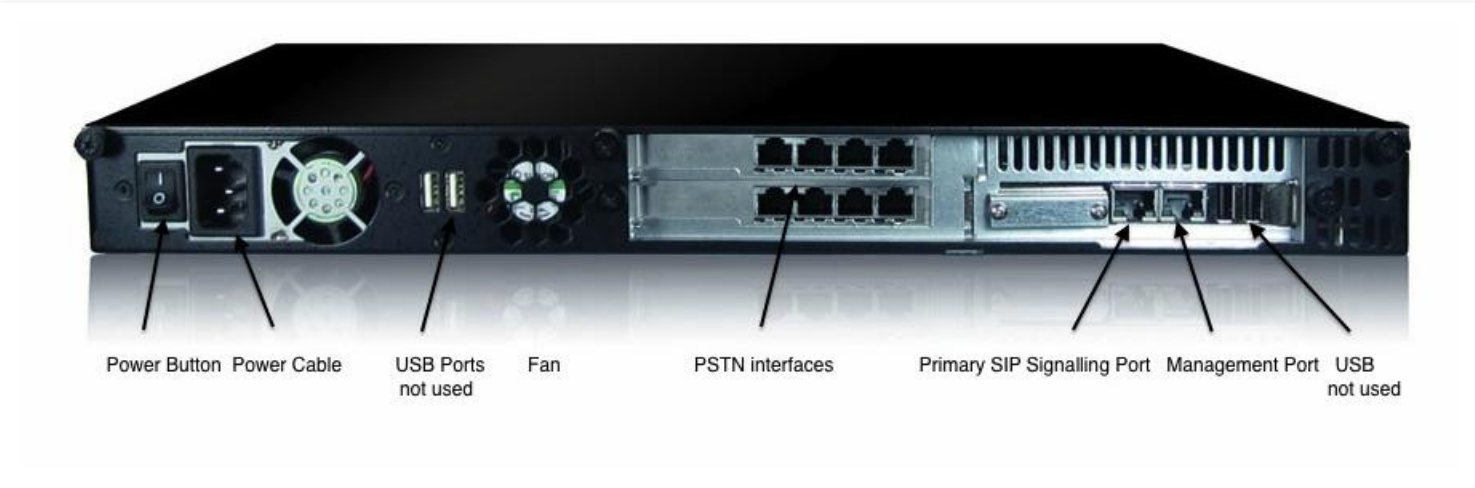
- Netborder SS7 to VoIP Appliance
  - Appliance can be 1U or 2U depending on model ordered
- Power Cable
  - AC cable in case of AC PSU (black cable)
  - DC cable in case of DC PSU (RED & Black cable)
- Mounting Brackets
- Quickstart user guide

What is not included

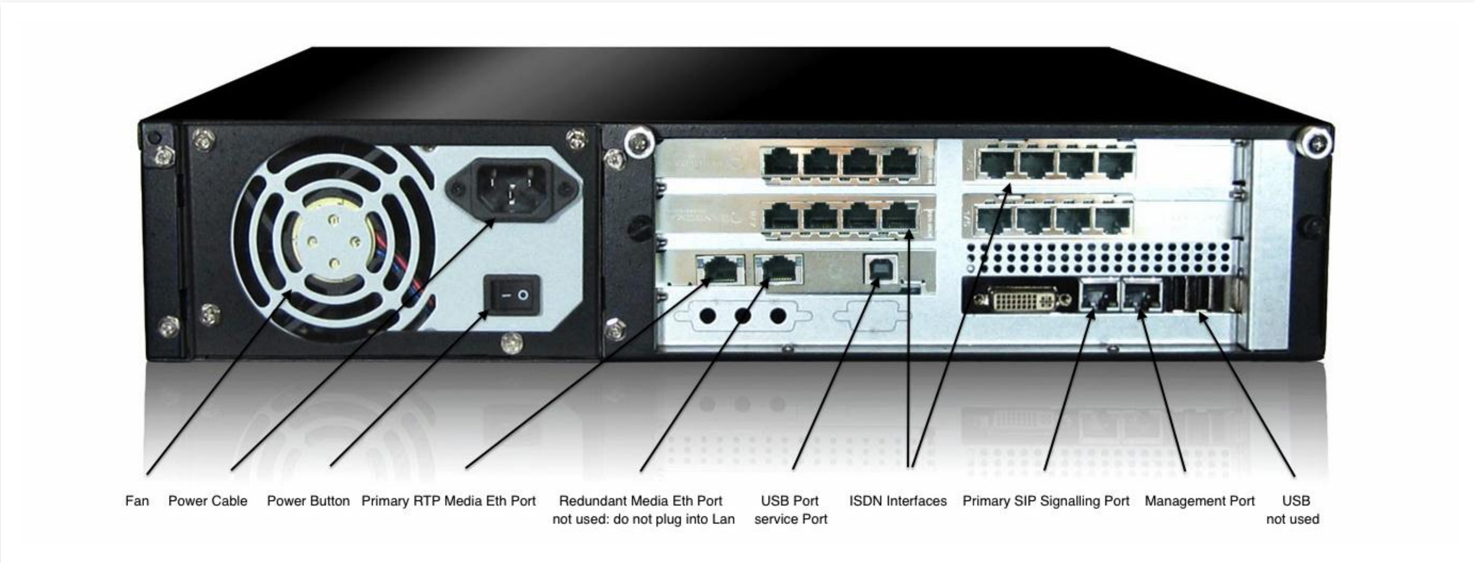
- Server Rails
  Server Rails are made based on standard. They can be purchased from any third party equipment vendor. For example: General Devices.

Front Panel

Rear Panel 1U



Rear Panel 2U



Rear Panel Description

- Fan
- Internal Power supply
  - Default AC, non-redundant
  - Option: DC or AC Redundant
- Power Button
- Unused Gig Ethernet Port
  - Not used at this time. Should NOT be plugged into the LAN.
- Primary Signaling and Media Gig Ethernet Port
  - This adapter must be plugged into the LAN
  - SIP Signaling and RTP Media will flow through this device.
  - WebUI identifies this device as "eth0"
- Secondary/Monitoring Gig Ethernet Port
  - This adapter is optional
  - It can be used for Monitoring and Statistics
  - WebUI identifies this device as "eth1"
- USB Ports
  - Used to re-flash the appliance
  - Future use: active/standby redundancy*

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

NSG Appliance Default Configuration

By default the NSG appliance gets shipped with following configuration.

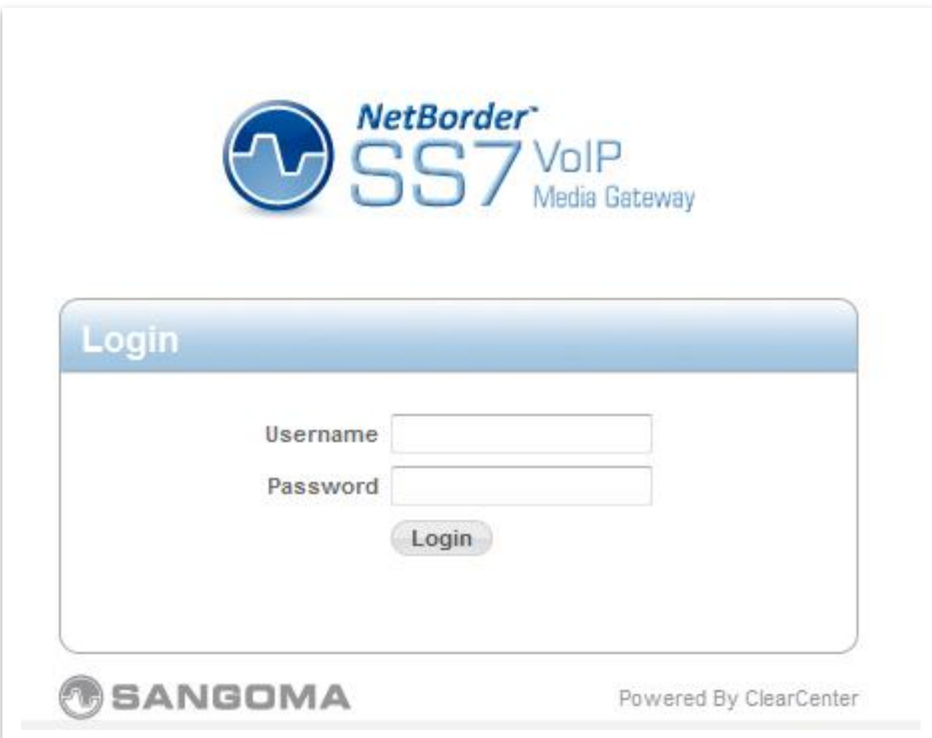
- Static IP 192.168.168.2
- Static IP Port eth0 (Primary SIP Signaling Port)

- WebUI URL http://192.168.168.2:81
- Username root
- Password sangoma

4. Initial Setup

Establishing a WebGUI Connection

- Connect Appliance Primary SIP Signaling Port to a LAN
- Configure Laptop or Desktop to IP address: 192.168.168.1
- Using a web browser connect to: http://192.168.168.2:81
- Login using above credentials

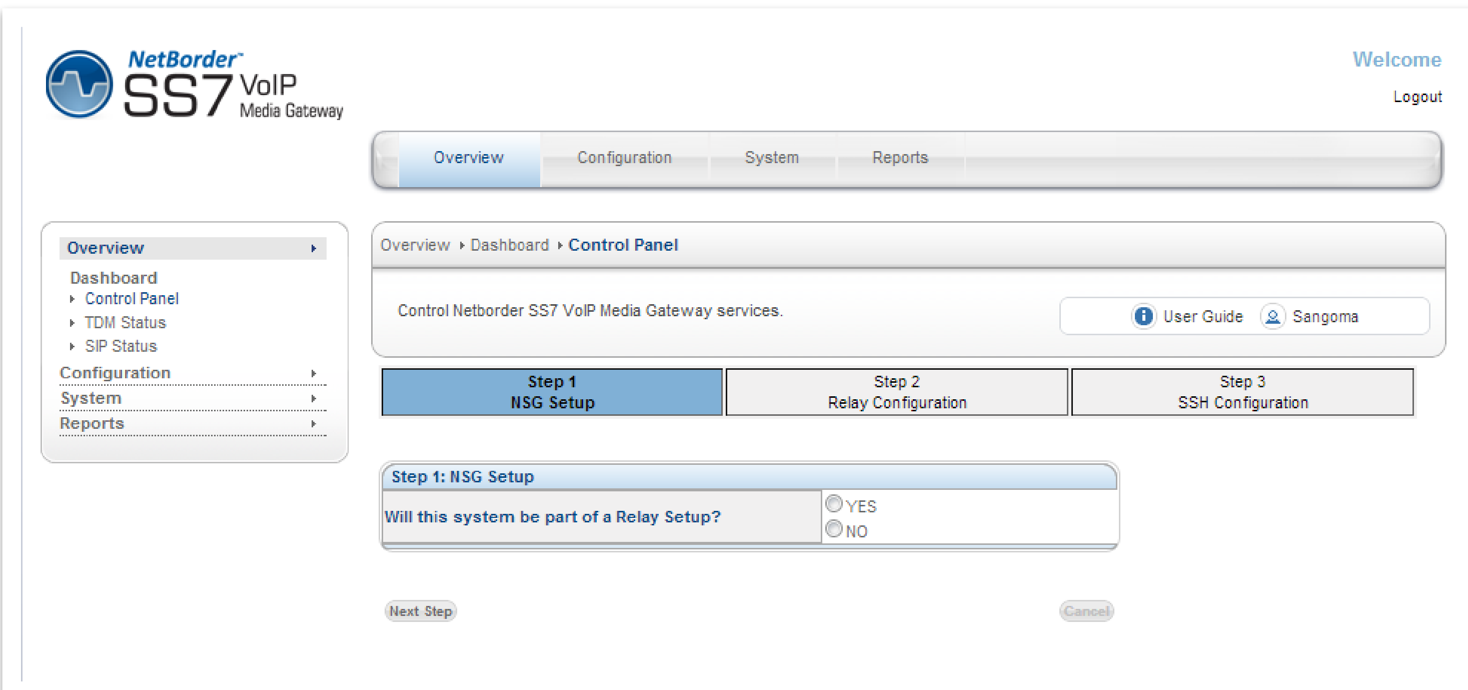


Relay Mode Check

ISUP Relay is a special feature of SS7 protocol that is used in distributed configurations. Distributed setups are based on Master/Slave configurations. Relay feature is fully described in “Relay“section of the document.

If configuring for SS7 ISUP Relay

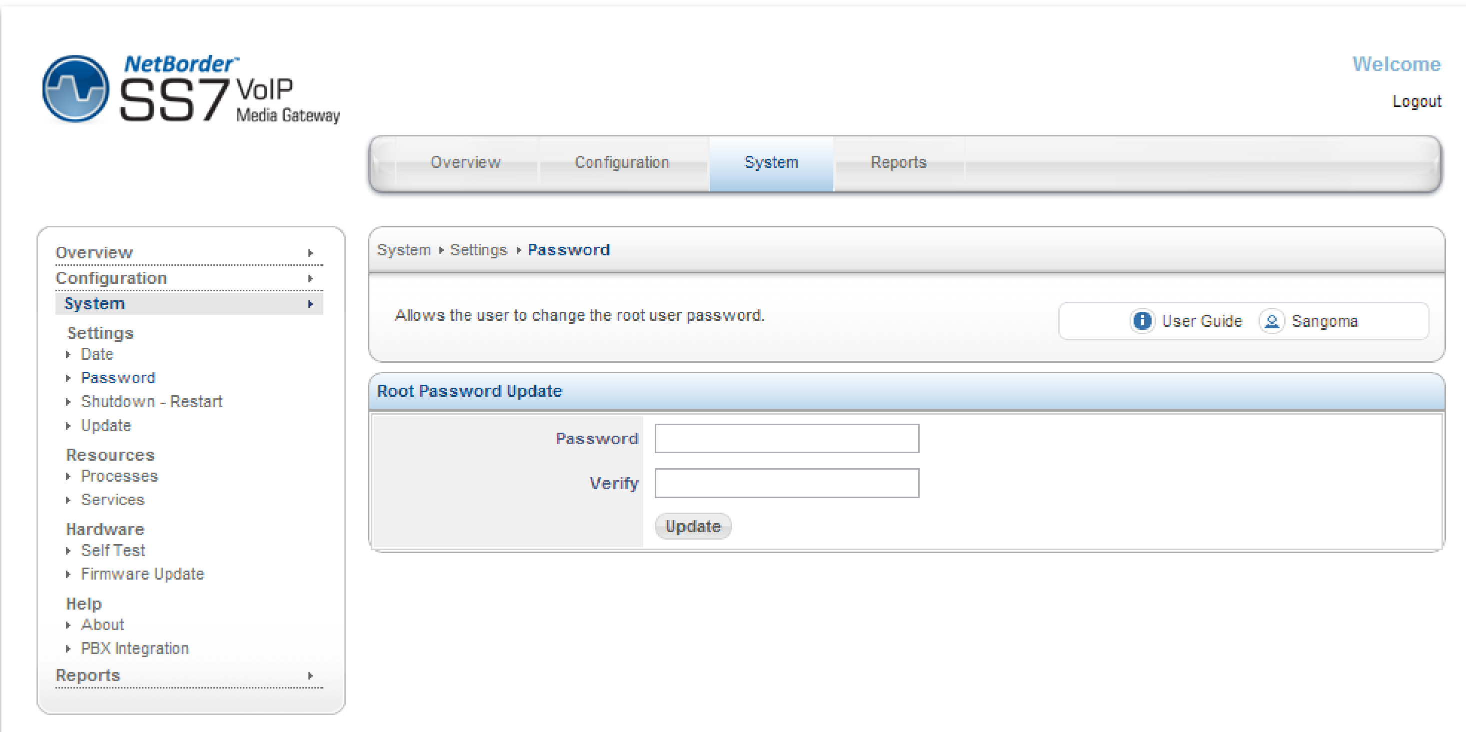
- Select Yes if this device will act as Master
- Select No for all other configurations. (Default)



Change Password

Sangoma NSG appliance comes with default password.
For security reasons please change the password.

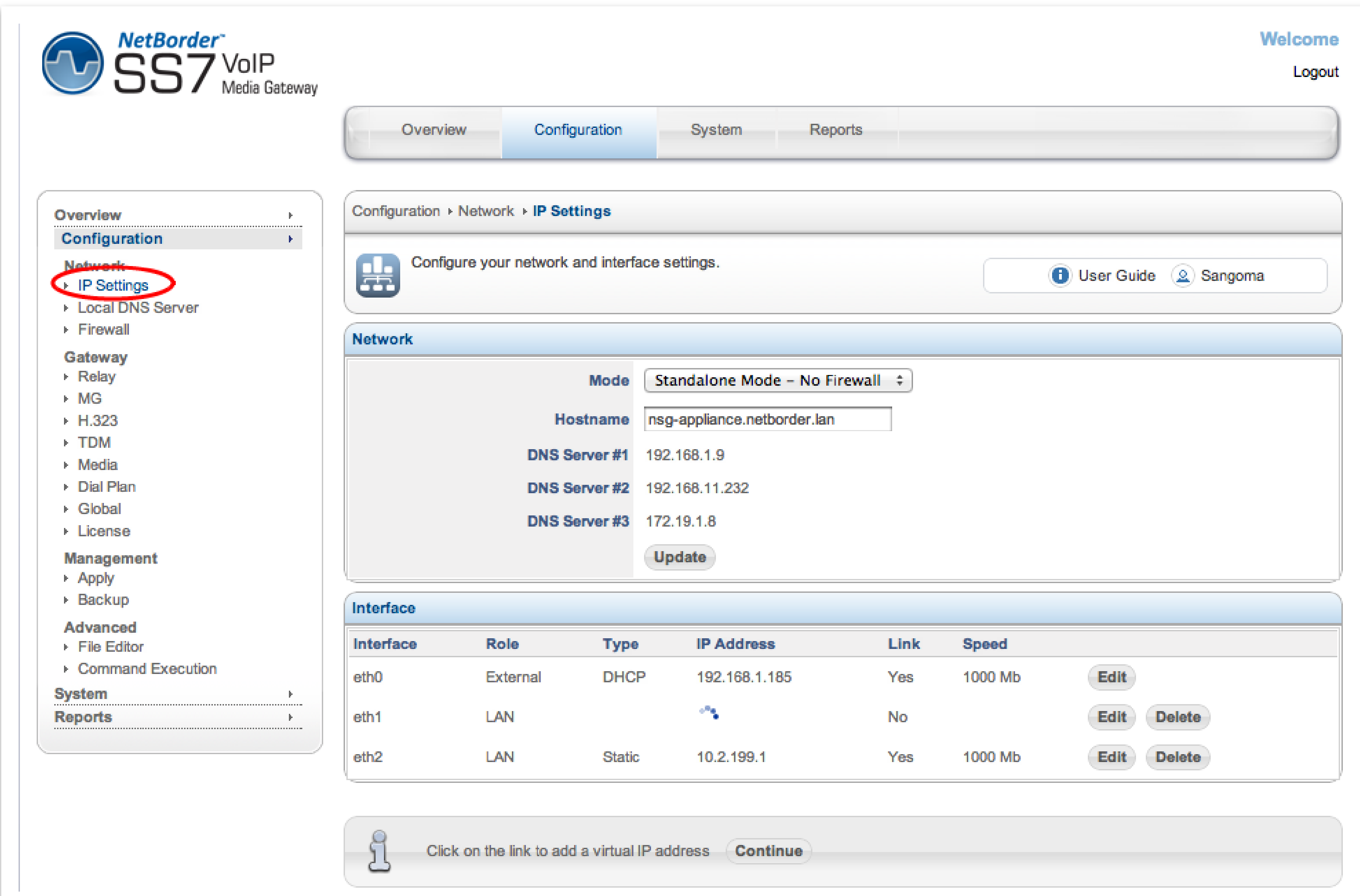
- Select **Password** page from side/top **System** menu
- Enter your new password
- Press update to save



Hostname & IP Address

By default the NSG appliance pre-configured with 192.168.168.2/24 address on Primary Port (eth0). The IP address can be changed based as follows

- Select **IP Settings** from side/top **Configuration** menu
- Specify Firewall Mode and Hostname
- Select **Edit** under eth0 and eth1 device and configure



NOTE
eth2 device is a Sangoma Transcoding device and should be modified.
Media section of the GUI will configure this device.

Appliance Network Interfaces

- eth0
  - Primary Signaling Port
  - By default provisioned as static 192.168.168.2
  - By default allows access to ssh and management http
- eth1
  - Secondary Signaling or Management Port
  - By default provisioned as static no IP address
  - By default allows access to ssh and management http
- eth2
  - Sangoma transcoding dsp board
  - Provisioned using Media page. Do not modify in this section.

Selecting Default Route

NSG appliance should have a single default route.
The default route is used to access Internet.
To configure a default route on eth0

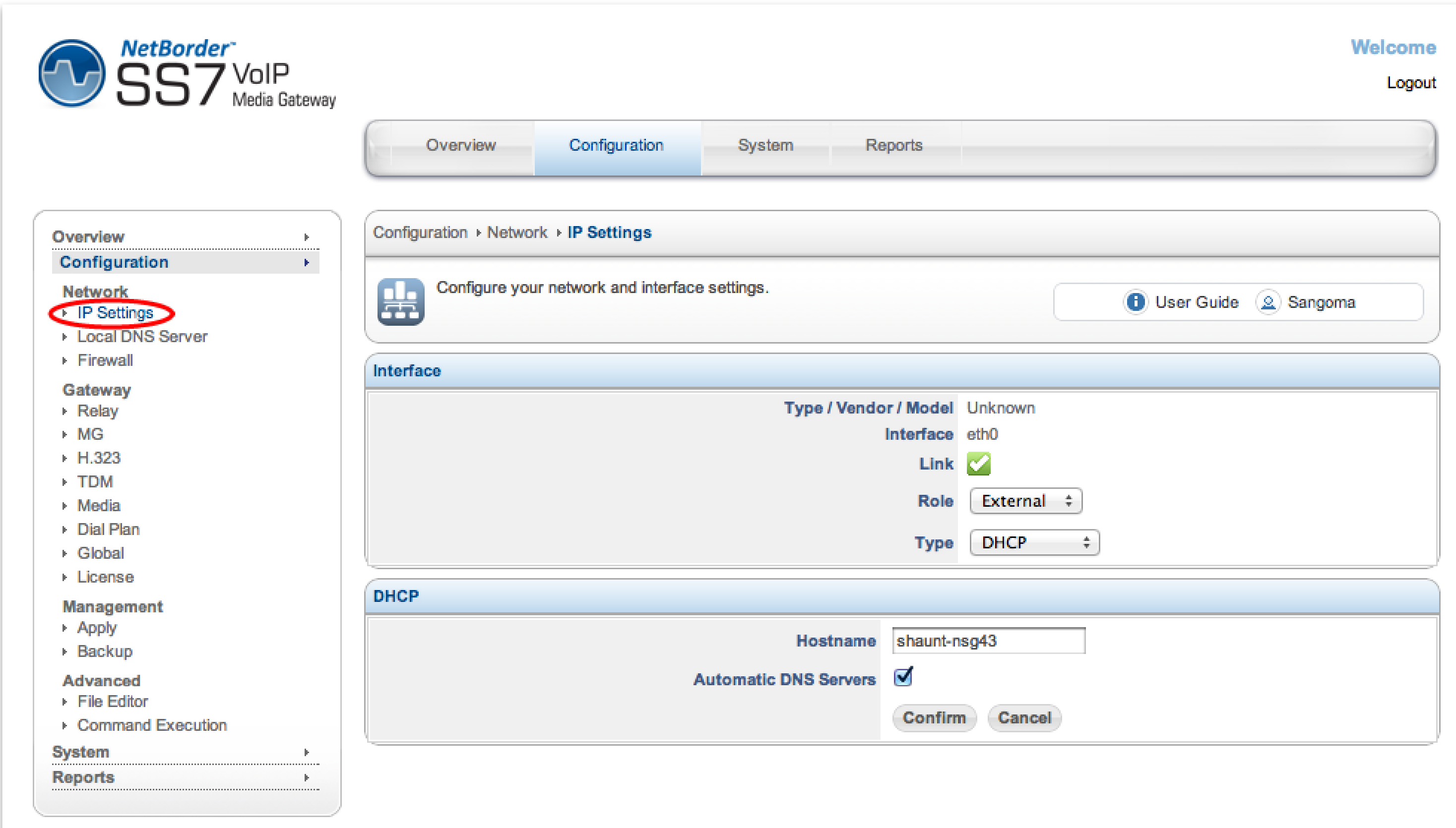
- Set the eth0 interface mode to External.
- Refer to section below.

Network Section

Variable Name	Input Options	Description
Mode	Standalone – No Firewall	Firewall Disabled
	Standalone	Firewall Enabled Warning: All active service ports must be explicitly enabled
Hostname	String	A hostname is the full name of your system. If you have your own domain, you can use a hostname like nsg.example.com Alternatively, you can also make one up: gateway.lan, mail.lan. The hostname does require at least one period (.)
Name/DNS Servers	Domain Name or IP address eg. 8.8.8.8	On DHCP and DSL/PPPoE connections, the DNS servers will be configured automatically for your IP Settings. In these two types of connections there is no reason to set your DNS servers. Users with static IP addresses should use the DNS servers provided by your Internet Service Provider (ISP). If you are using Multi-WAN, please review the documentation on the topic of DNS servers.

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

Interface Section



Network Role

When configuring a network interface, the first thing you need to consider is the network role in IP Settings. Will this network card be used to connect to the Internet, for a local network, for a network with just server systems? The following network roles in IP Settings are supported in NSG and are described in further detail in the next sections:

- External - network interface with direct or indirect access to the Internet
- LAN - local area network
- Hot LAN - local area network for untrusted systems
- DMZ - de-militarized zone for a public network

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

Option	Description
External	Network interface with direct or indirect access to the Internet External interface is used as the system default route. WARNING: You should have only ONE external network interface. Usually eth0 is the external interface
LAN	Connection to your local network Usually eth1 is the LAN interface
Hot LAN	Hot LAN (or “Hotspot Mode”) allows you to create a separate LAN network for untrusted systems. Typically, a Hot LAN is used for: • Servers open to the Internet (web server, mail server) • Guest networks • Wireless networks A Hot LAN is able to access the Internet, but is not able to access any systems on a LAN. As an example, a Hot LAN can be configured in an office meeting room used by non-employees. Users in the meeting room could access the Internet and each other, but not the LAN used by company employees.
DMZ	In NSG, a DMZ interface is for managing a block of public Internet IP addresses. If you do not have a block of public IP addresses, then use the Hot LAN role of your IP Settings. A typical DMZ setup looks like: • WAN: An IP addresses for connecting to the Internet • LAN: A private network on 192.168.x.x • DMZ: A block of Internet IPs (e.g from 216.138.245.17 to 216.138.245.31) NSG GUI has a DMZ firewall configuration page to manage firewall policies on the DMZ network.

Types

Option	Description
DHCP	For most cable and Ethernet networks, DHCP is used to connect to the Internet. In addition, your system will have the DNS servers automatically configured by your ISP when the Automatic DNS Servers checkbox is set.
Static	If you have a static IP, you will need to set the following parameters: • IP • Netmask (e.g. 255.255.255.0) • Gateway (typically ends in 1 or 254)
PPPoE DSL	For PPPoE DSL connections, you will need the username and password provided by your ISP. In addition, your system will have the DNS servers automatically configured by your ISP when the Automatic DNS Servers checkbox is set.

Virtual IP's

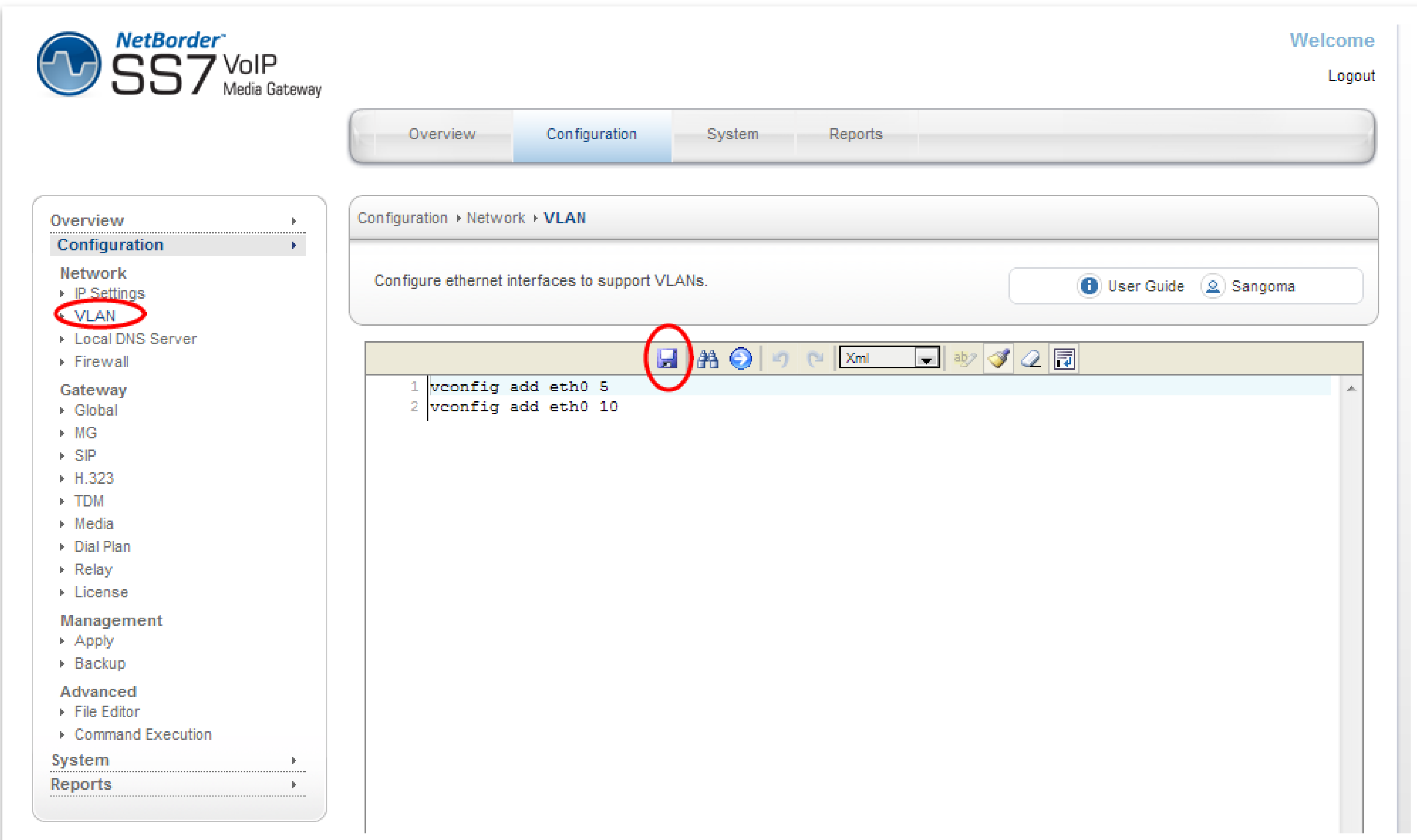
NSG supports virtual IPs. To add a virtual IP address, click on the link to configure a virtual IP address and add specify the IP Address and Netmask. You will also need to create advanced firewall rules if the virtual IP is on the Internet.

VLAN

VLAN Configuration

Currently NSG only supports VLAN configuration via CLI/SSH or GUI File Editor

- Select **VLAN** from side/top **Configuration** Menu
- Copy in the VLAN configuration script
- Save



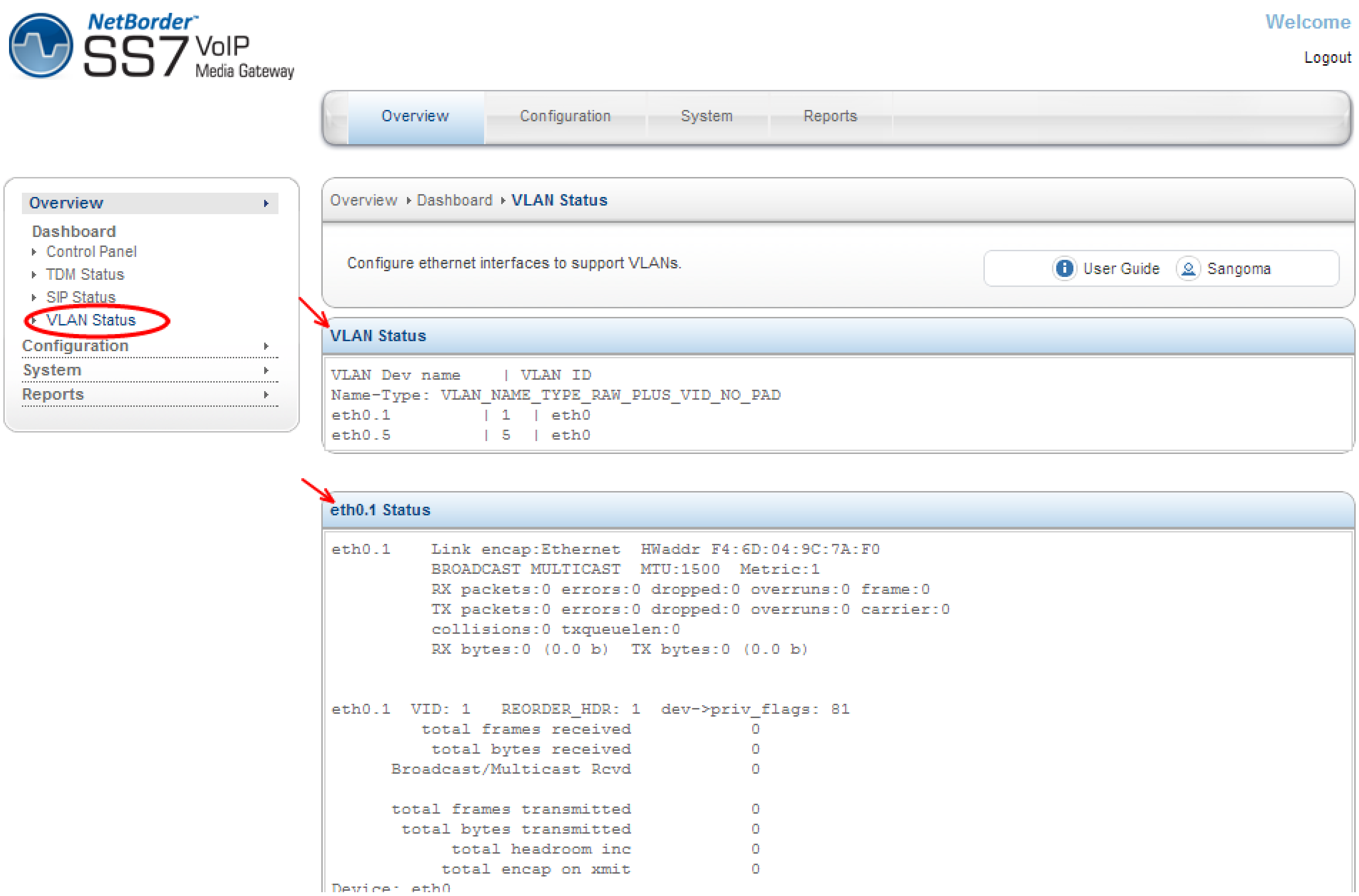
100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

Sample Script that is to be copied into the VLAN config startup script.

```
#Create a VLAN device on eth0 interface  with VLAN ID of 5
vconfig add eth0 5

#Configure VLAN device with IP/Netmask
ifconfig eth0.5 192.168.1.100 netmask 255.255.255.0 broadcast 192.168.1.255 up

#List of all configured VLAN devices
cat /proc/net/vlan/config

#Statistic for specific vlan device
cat /proc/net/vlan/eth0.5

#Remove VLAN device
ifconfig eth0.5 down
vconfig rem eth0.5
```

VLAN Status

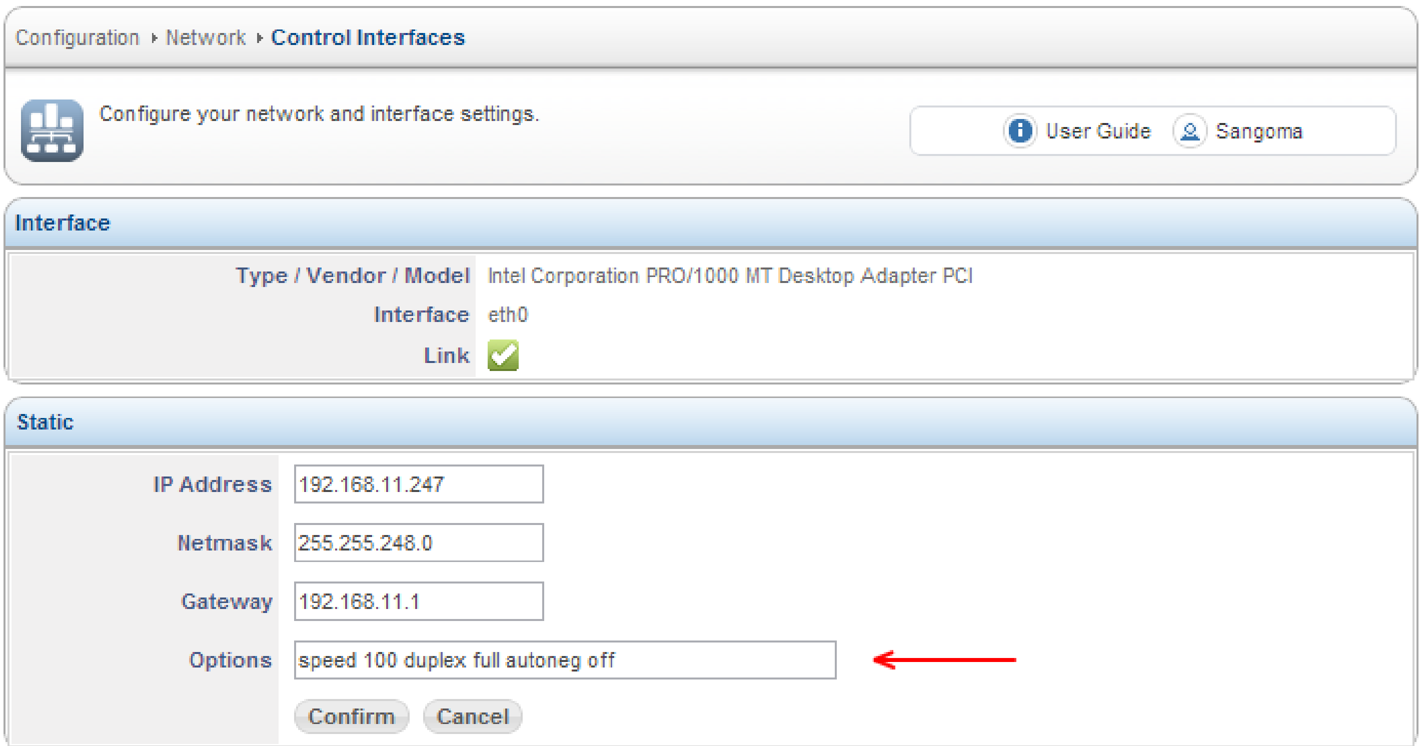
- Select **VLAN** Status from side/top **Overview** Menu
- This page shows
    - All configured VLANs
    - Individual VLAN configuration
    - Individual VLAN IP information



Ethernet Options

Setting custom Ethernet options such as disabling auto negotiation is done as part of the IP Settings must be done via CLI or WebGUI File Editor.

- Select **IP Settings** from side/top **Configuration** Menu



Specify **Options** field in order to add special configuration to this interface.

IP Troubleshooting

In most installs, the network cards and IP settings will work straight out of the box. However, getting the network up the first time can be an exercise in frustration in some circumstances. Issues include;

- Network card compatibility
- Invalid networks settings (username, password, default gateway)
- Cable/DSL modems that cache network card hardware information

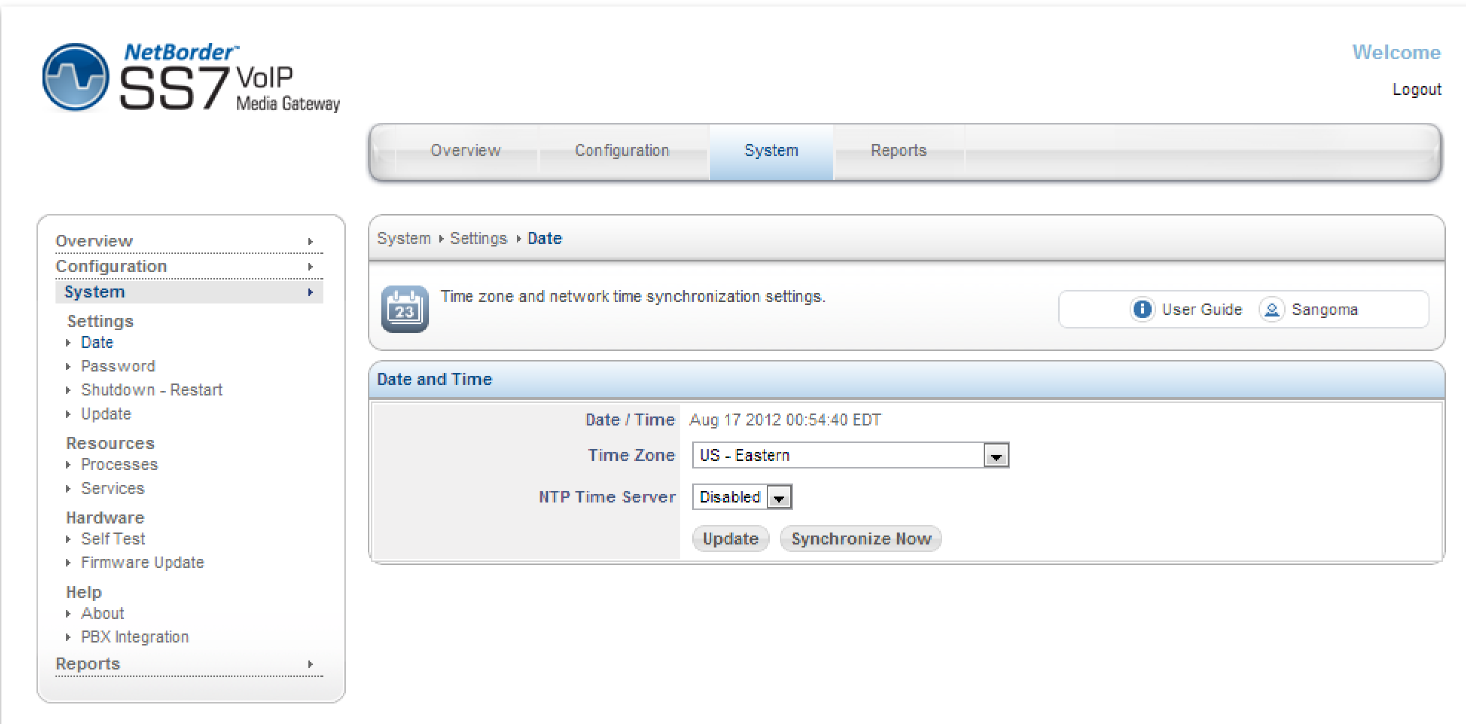
Date & Time Service Config

The Date/Time configuration tool allows you to:

- Select your time zone
- Synchronize your clock with network time servers
- Enable/disable a local time server for your network

To configure

- Select **Date** from side/top **System** menu
- Refer below to all available options.



Option	Description
Date/Time	The system date, time and time zone information is displayed for informational purposes. Please make sure it is accurate since it is not unusual to have computer clocks improperly set on a new installation.
Time Zone	It is important to have the correct time zone configured on your system. Some software (notably, mail server software) depends on this information for proper time handling.
NTP Time Server	An NTP Time Server is built into NSG.
Time Synchronization	Hitting the Synchronize Now button will synchronize the system's clock with network time servers.

Self Test

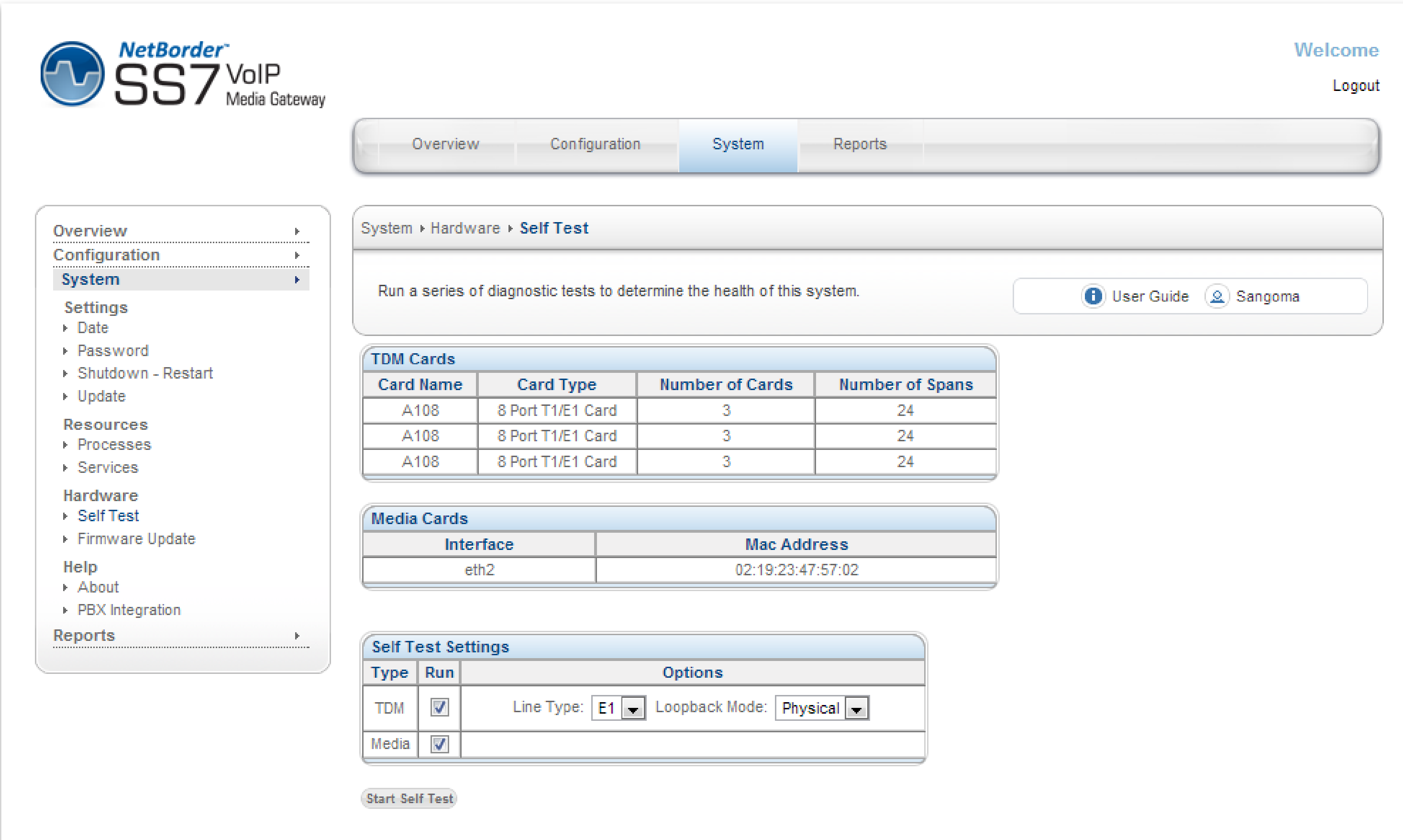
Self-Test page must be run on initial installation or on any hardware upgrade. It will run a battery of tests on Sangoma TDM and Transcoding hardware.

WARNING:

- Self-Test will change your configuration and restart the service.
  All services during the Self-Test will be stopped.
- Do not run Self-Test in production!
- Only run Self-Test during on initial setup or during a maintenance window.
- The existing configuration will not be affected.

The Self-Test can be used to detect:

- Defective TDM hardware
- Defective Media Transcoding hardware
- Miss-configured system device drivers
- PCI Interrupt errors
- Motherboard System issues



Running Self-Test

- Select Self Test from side/top System Menu
- If in North America select T1
- If not in North America select E1
- Select Media Transcoding Hardware if present.
- Click Start Self-Test

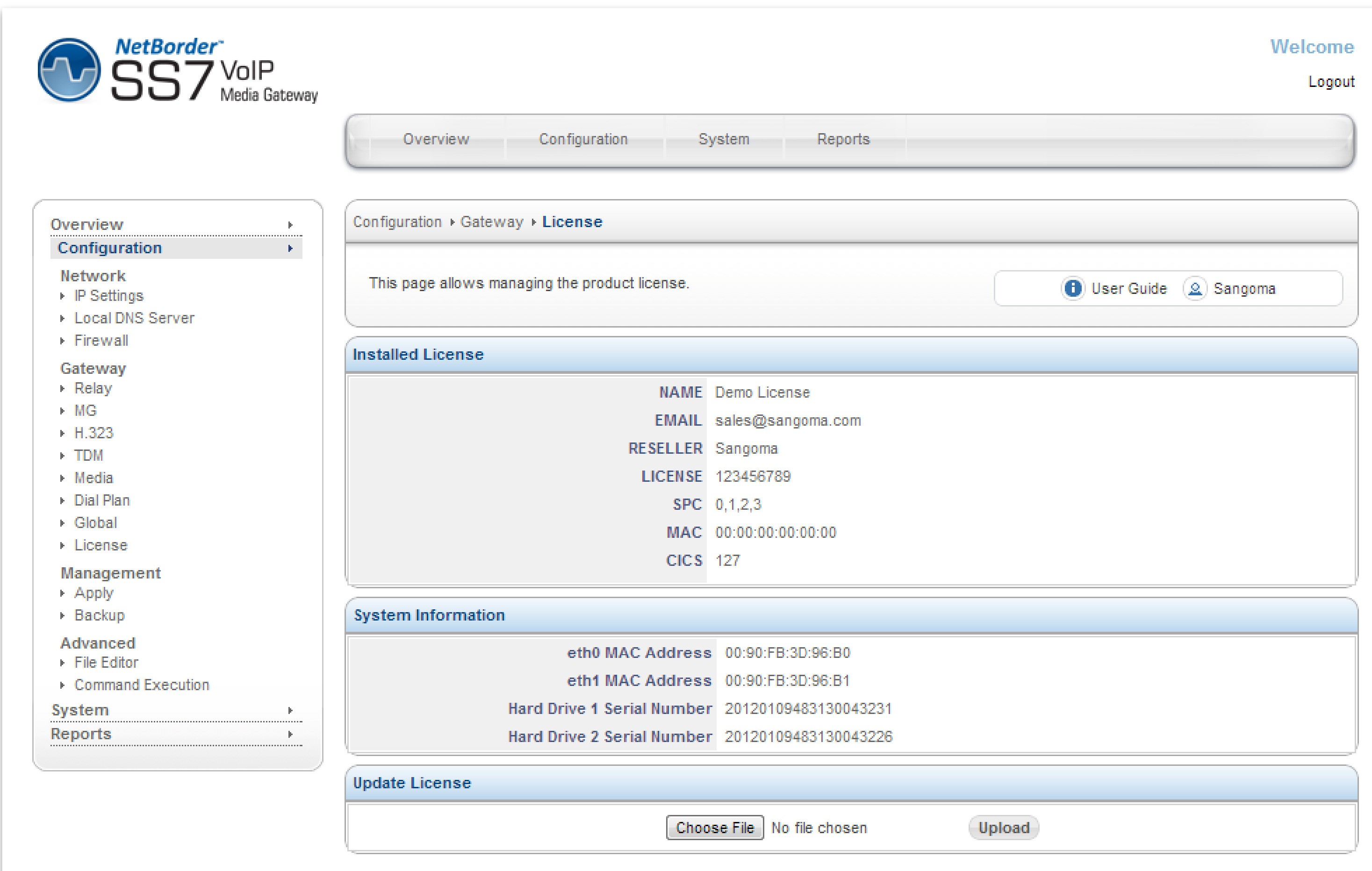
NSG License

To update NSG license

- Select **License** from side/top **Configuration** Menu

- Obtain NSG License from Sangoma Support
- Upload the License into the NSG Gateway via the **Upload** Button

The License page offers the detailed license overview. Each NSG appliance comes with pre-installed license. In case of upgrades, of expansions please contact Sangoma Sales.

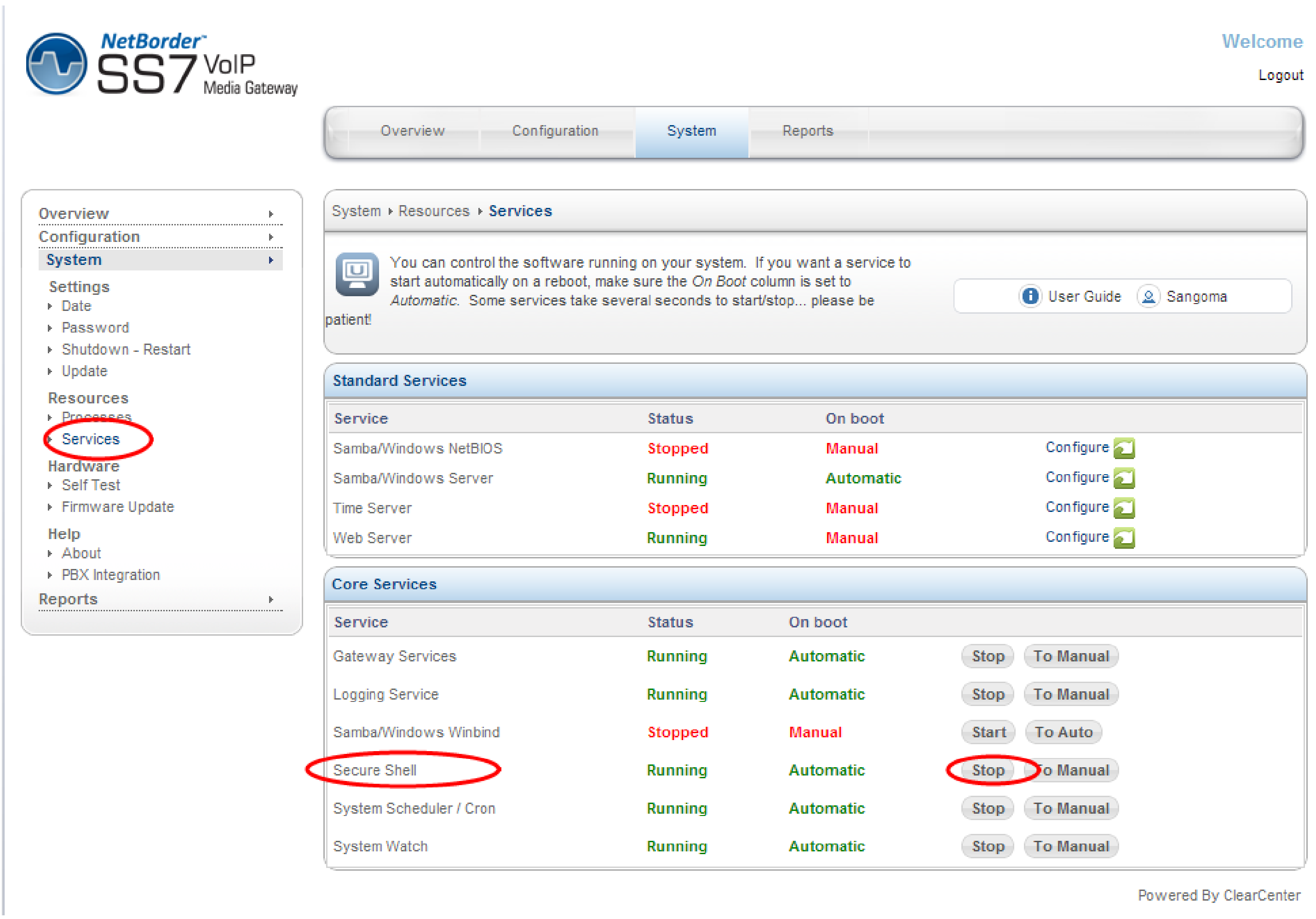


Console SSH Configuration

By default NSG systems come with SSH **enabled**.

To configure ssh service

- Select **Services** from side/top System Menu
- Enable or disable **Secure Shell** service



5. User Interface

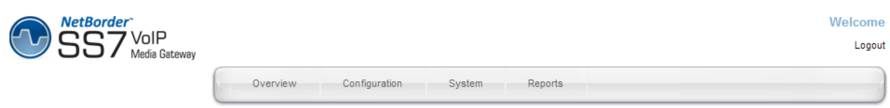
Netborder SS7 to VoIP media gateway provides the user with two interfaces

- WebGUI
  - Web GUI is preferred for almost all operations
  - Configuration, Operations, Statistics, Reports

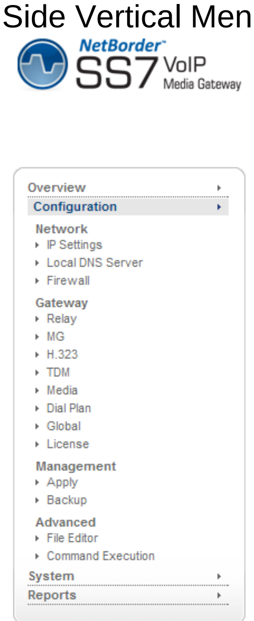
- Console (ssh)
  - For power users familiar with Linux operating system, ssh console provides advanced and flexible interface for troubleshooting and automation.

WebGUI

- WebGUI resides on the port **81**
- Interface provides two identical menus for easy access to all options
  - Top Horizontal Menu



  - Side Vertical Menu



WebGUI Structure

Overview

- Control Panel
    - Used to control the gateway operations: start,stop
- TDM Stats
    - Provides full overview of gateway utilization and states
- SIP Stats
    - Provides full SIP statistics, call count
- VLAN Stats
    - Provides full VLAN statistics, VLAN ID, IP, Netmask for each VLAN.

Configuration

- Network
    - Allows network configuration such as IP, VLAN, DNS and Firewall
- Gateway
    - Core product configuration
    - Provides configuration of all Signaling and Media Protocols
        - SIP, RTP,H323, Media Processing, Megaco(MG), SS7/Sigtran (TDM)
    - Routing Logic / Dialplan
        - XML based dialplan
- Management
    - Apply
        - Write all configurations changed and set in Gateway section.
    - Backup
        - Backup all system configurations into a zip file.
        - Recover a system from a backup file
- Advanced
    - File Editor
        - Allows custom file editing for custom configuration
        - Troubleshooting
    - Command Execution
        - Instead of logging into a shell
        - Execute any system command via the WebGUI.

System

- Settings
    - Date
        - Set date time and sync to time server
    - Password
        - Change password
    - Shutdown
        - Shutdown or reboot a system

  - Update
    - Software and patch update system
- Resources
  - Processes
    - List of currently running process
  - Services
    - List of all available services
    - SSH service start/stop
- Hardware
  - Self-Test
    - Allow for system software and hw components test.
  - Firmware Update
    - Allows for firmware updates
    - Sangoma TDM boards
    - Sangoma Media processing boards
- Help
  - About
    - Shows system version and version of all important packages.
  - PBX Integration
    - Help documentation

Reports

- Dashboard
  - Overview
    - Overview of network interfaces
- Network
  - Network Report
    - Long term usage charts for each network device
  - Protocol Capture
    - PCAP packet capture with filter support for any network interface
- System
  - Gateway Logs
    - Specific gateway logs used to quickly trouble shoot gateway issues
    - Allows for log download
  - Advanced Logs
    - Full system wide logs with filters
  - Hardware Report
    - Full hardware overview and description
    - HDD, Memory and system usage
    - Device enumeration
  - Resource Report
    - Long term statistics

Console Structure

- Console access via ssh
- Operating system is Linux based. Therefore Linux expertise is mandatory.
- WARNING
    - Working in shell is very powerful and flexible, but also dangerous
    - A system can be corrupted, formatted, erased if user makes a mistake.

System Commands

All system commands are based on Linux operating systems. Listed here are some most useful debugging commands.

- tcpdump
    - Provides network capture to a pcap file
- ethtool
    - Provides detail network interface information, like Ethernet link status.
- Ifconfig
    - Network interface statistics tool
    - Shows error counters on Ethernet and TDM interfaces.
- wanpipemon
    - Sangoma TDM troubleshooting tool

Refer to the appendix for all System Commands

Gateway CLI

Once in the ssh shell, user can enter the NSG gateway CLI.
NSG Gateway CLI offers advanced gateway statistics and troubleshooting. As well as detail protocol and channel statistics.

- status
    - Call statistics summary of: active calls, calls per sec, total calls
- show channels
    - Detailed list of current active calls
- mg <enter>
    - Megaco statistics
- ftdm ss7 <enter>
    - SS7 statistics

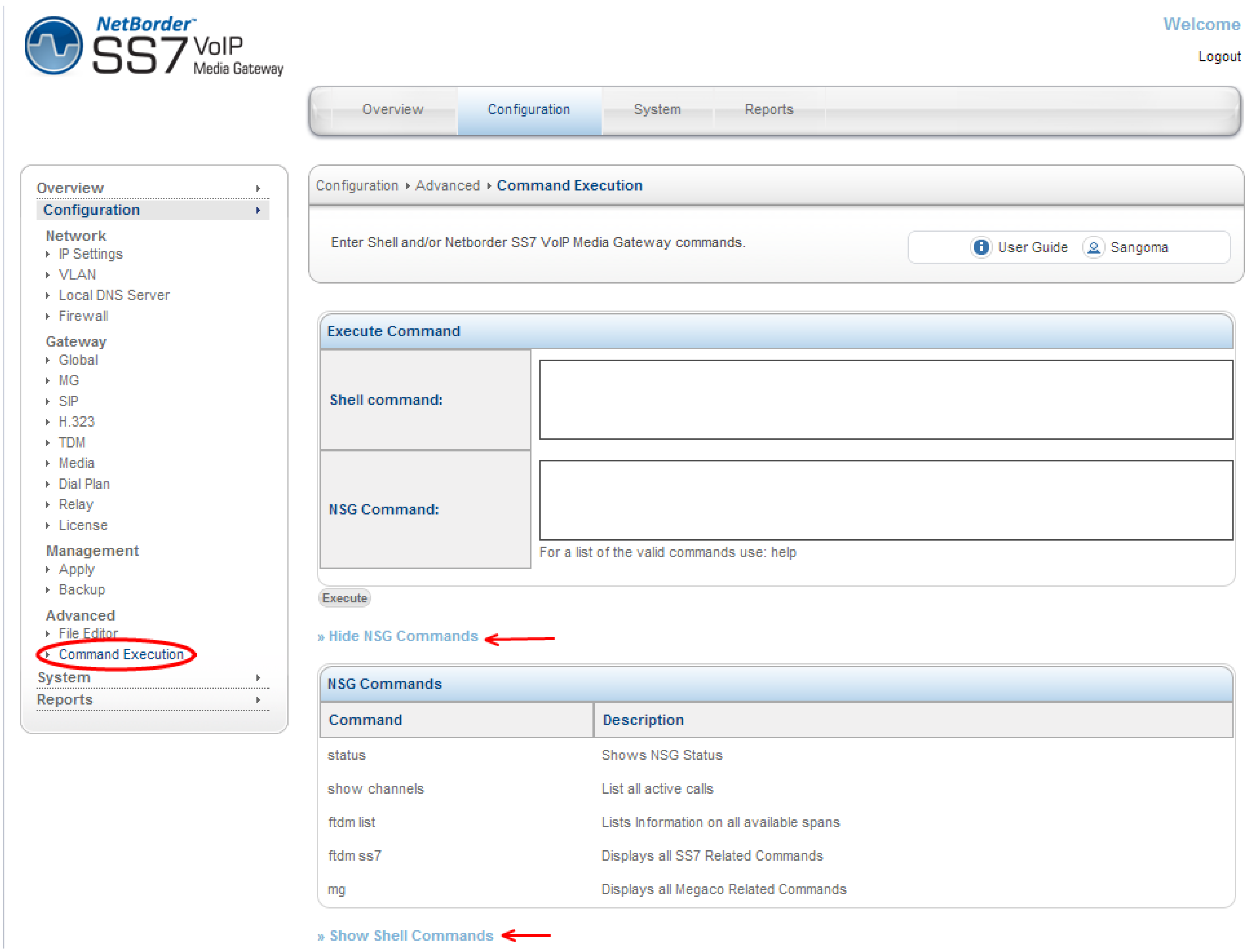
100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

- ftdm list
    - list of all tdm channels

Please refer to the appendix for all Gateway CLI commands.

Shell/CLI from GUI

- Select **Command Execution** from side/top **Configuration** Menu
- Specify a shell or CLI command. Refer to guide below.



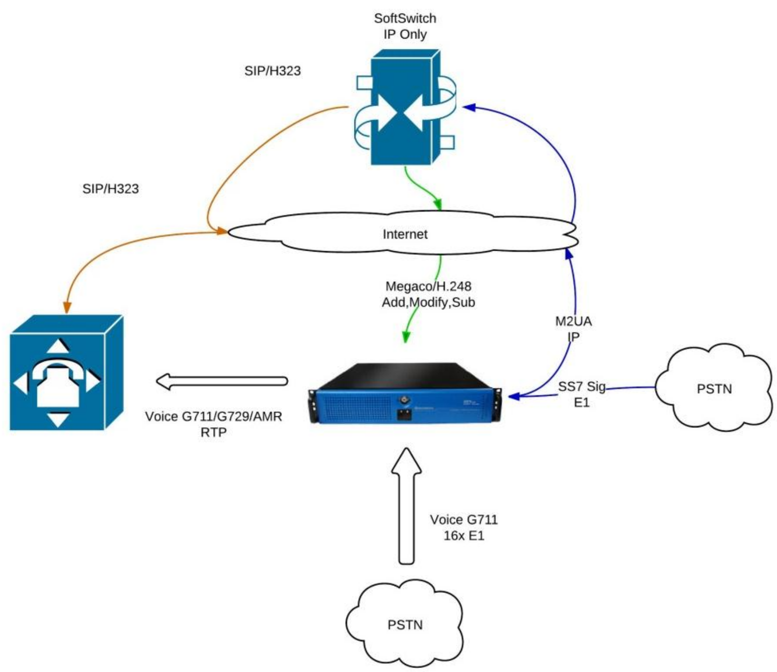
6. Usage Scenarios

Signaling Gateway: M2UA

- Pass through signaling from TDM to IP
  - MTP2 -> M2UA
- Pass through signaling from IP to TDM
  - M2UA -> MTP2

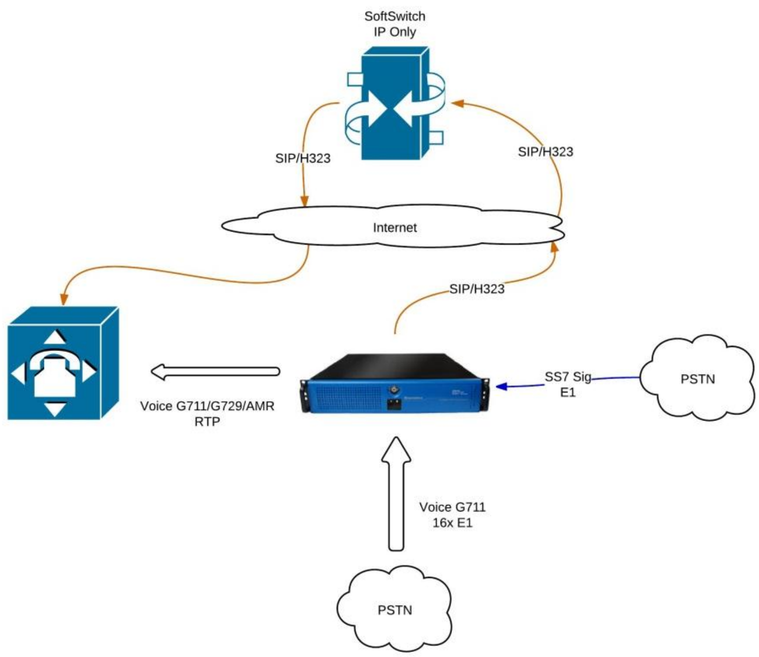
Megaco/H.248 Media Gateway: MG + SG

- Third part Softswitch/MGC controlling Netborder SS7 Media Gateway using Megaco/h.248 protocol.
  - Bridge RTP media to TDM Voice 64kb G711 channels
  - Bridge TDM Voice 64kb G711 channels to RTP media ports
- Media specific functions
  - Transcoding
  - DTMF
  - T38 Faxing



SIP/H323 to SS7 ISUP

- Bridge signaling sessions from H323 to SS7 ISUP
  - Bridge RTP media to TDM Voice 64kb G711 channels
- Bridge signaling session from SS7 ISUP to H323
  - Bridge TDM Voice 64kb G711 channels to RTP media ports
- Media specific functions
  - Transcoding
  - DTMF
  - T38 Faxing



Any to Any Signaling and Media Gateway

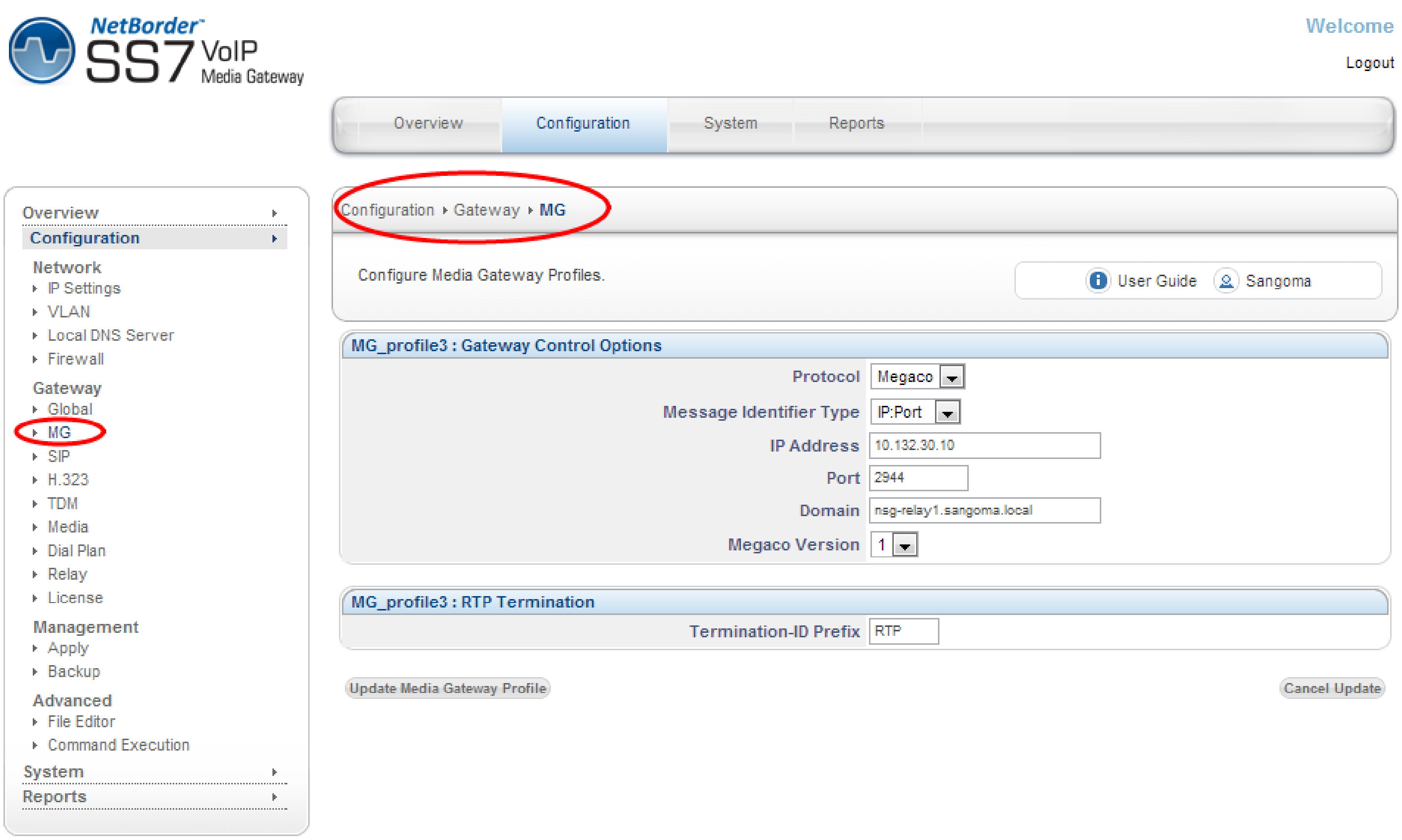
- Route any signaling traffic from eny signaling endpoint simultaneously.
- Ability to run all protocols together at the same time.
- Route media with transcoding/dtmf/T.38 to/from end media endpoint.

7. Megaco/H.248 Media Gateway Configuration

Create MG Profile

Media gateway profile will contains all the required configuration parameters to bring up the Media gateway stack.

- Select **MG** from the side/top Configuration menu
- Select **Add New Profile**
  - Use default profile name, or specify one
- Select **Create Media Gateway Profile**
- Configure the MG Profile based on information received from our provider.
- Select **Update Media Gateway** Profile to save



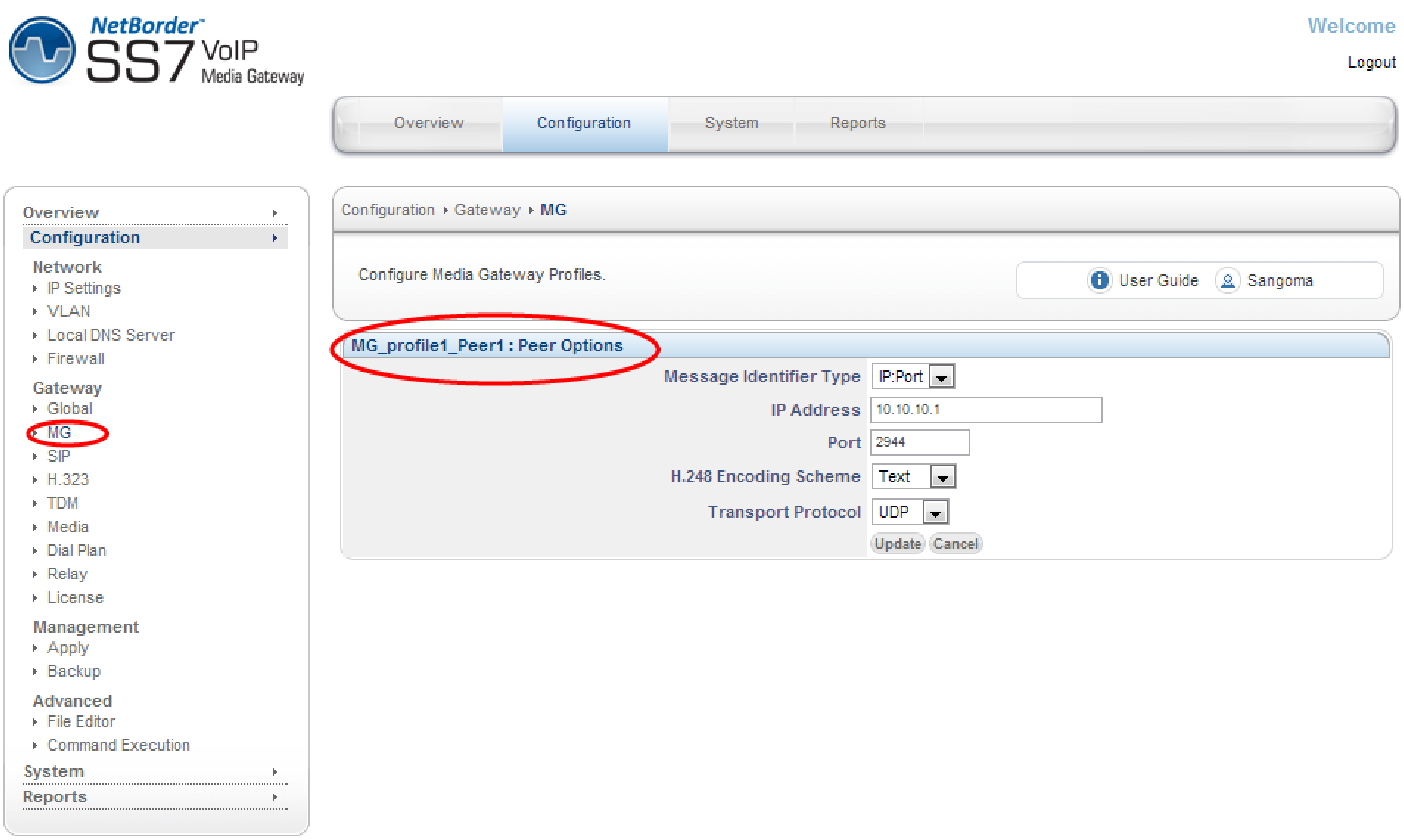
Followings are the fields which need to to be configured.

Field Name	Possible values	Default Values	Description
Protocol	MEGACO MGCP	MEGACO	Type of protocol Media Gateway is going to use. NOTE: Currently Media Gateway supports only MEGACO
Message Type Identifier	IP-PORT IP DOMAIN	IP-PORT	Media gateway message identifier (MID) type field will be used to build the message identifier field which Media Gateway will used in all the originating messages. For example: If MID type is IP-PORT then Message identifier format will be “[IP-Address]:Port” If MID type is DOMAIN then message identifier format will “<Domain>” . Refer to Domain section below. If MID type is IP then message identifier format will “[IP-Address]” Note: IP-Address, Port and Domain values will be as defined above.
IP Address	any ipv4 addr	NA	Media Gateway source IP address.
Port	1 - 65000	NA	Media Gateway source Port.
Domain	(a string value)	NA	Media Gateway domain name. Used as MID Type, when MID Type is set to DOMAIN. Ignored if MID Type is not Domain. Default to system domain name.
Megaco Version	1 2 3	1	Megaco protocol version which Media Gateway will use while communicating with Media Gateway Controller
Termination-ID Prefix	any number starting from 1	NA	RTP termination id prefix which Media Gateway will use while allocating RTP terminations. This variable is used as a name of RTP termination. Eg: RTP/1, RTP/2 …

Create MG Peer Profile

Each Media gateway profile will associate with one or multiple peers.
NOTE: As of now NSG supports only “one peer per MG profile”.

- Select **Add Peer** in MG Section
- Fill in the peer information
- Select **Update** to Save



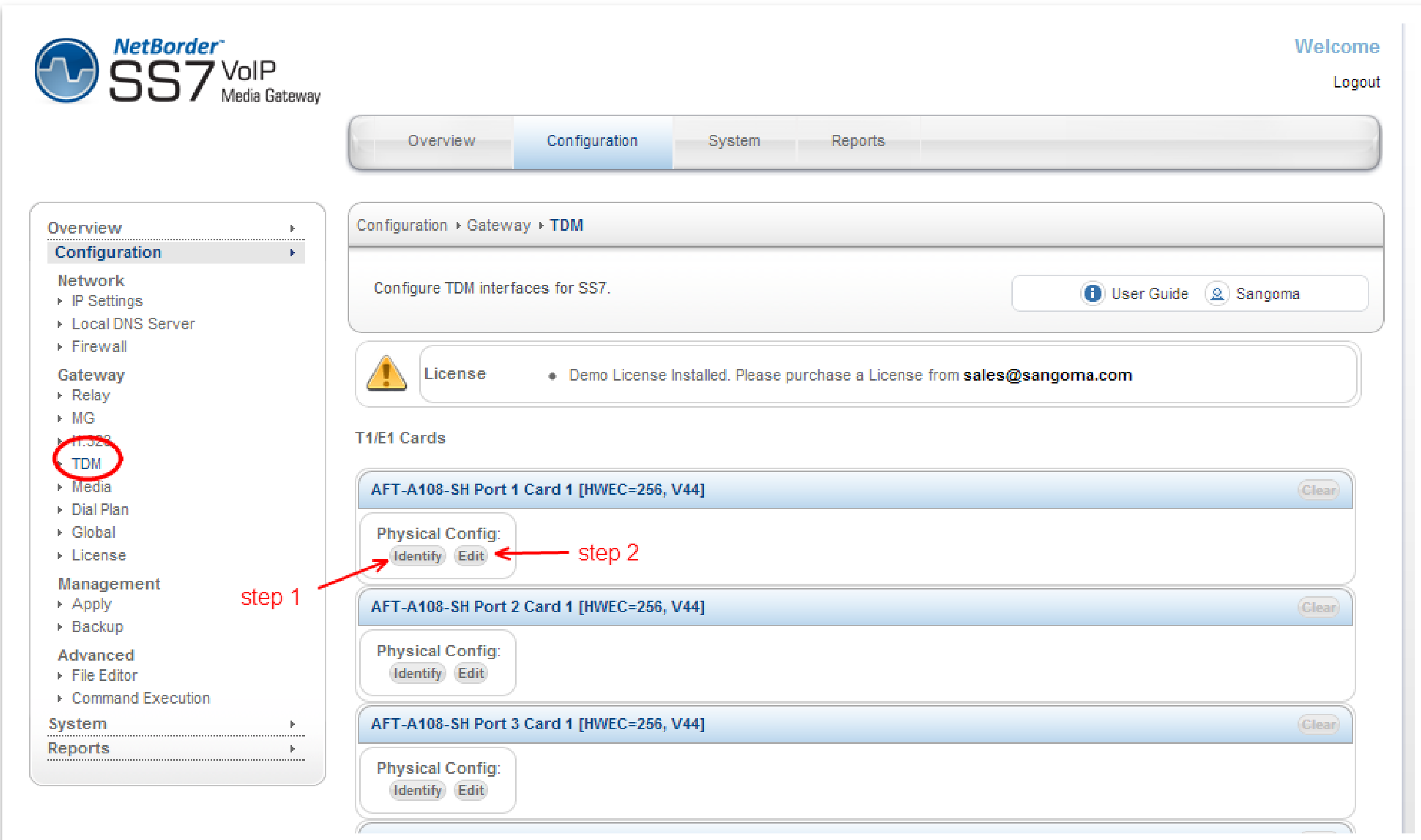
Followings are the fields which need to be configured.

Field Name	*Possible values*	*Default Values*	*Description*
Message Identifier Type	IP-PORT IP	IP-PORT	Media gateway Controller message identifier (MID) type field will be used by Media Gateway to identify the peer. Message identifier value will be built based on MID type field. For example: If MID type is IP-PORT then Message identifier format will be "[IP-Address]:Port" If MID type is IP then message identifier format will "[IP-Address]" Note: IP-Address and Port values will be as defined above.
IP Address	NA	NA	Media Gateway Controller IP address.
Port	NA	NA	Media Gateway Controller Port number
H.248 Encoding Scheme	TEXT BINARY	TEXT	Encoding scheme of MEGACO protocol which will be used by Media Gateway while encoding/decoding the H.248 messages.
Transport Protocol	UDP TCP SCTP	UDP	Media Gateway will use the transport type field to decide which transport to use for transmitting/receiving MEGACO messages. NOTE: currently we are supporting only UDP/TCP.

- Once the **Media Peer** is configured the Megaco configuration section is complete.
- Proceed to **TDM Termination for Media Gateway**

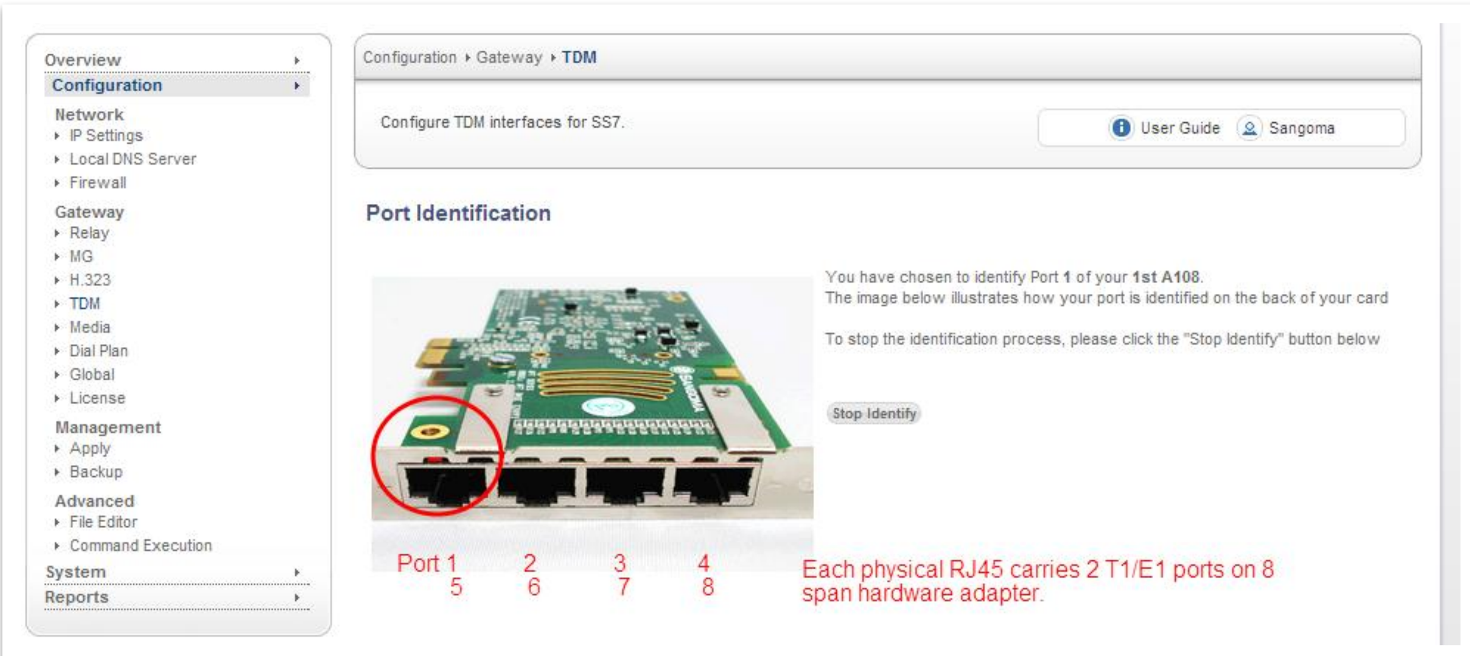
TDM Termination for Media Gateway

- Select **TDM** from side/top **Configuration** menu
- The TDM section will display all installed TDM Spans/Ports.



Identify

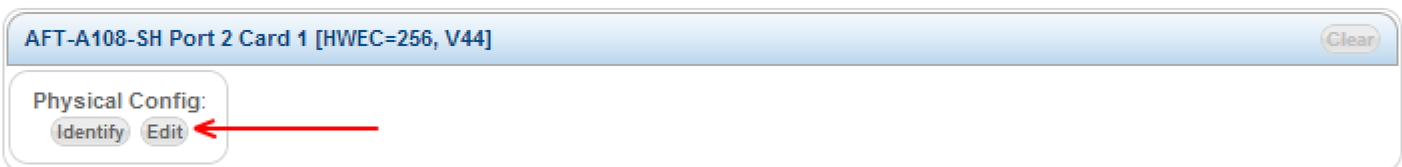
- In order to determine which physical T1/E1 port is: Port 1 Card 1
- Select **Identify** button for Port 1 Card 1
- The LED light will start flashing on a rear RJ45 T1/E1 port: rear panel.
- Look at the rear panel of the appliance and plug in RJ45 cable to the blinking RJ45 T1/E1 port.
- Once the Port 1 Card 1 is identified, the subsequent ports for that board are labeled.
- Or alternatively keep using the Identify feature for each port.



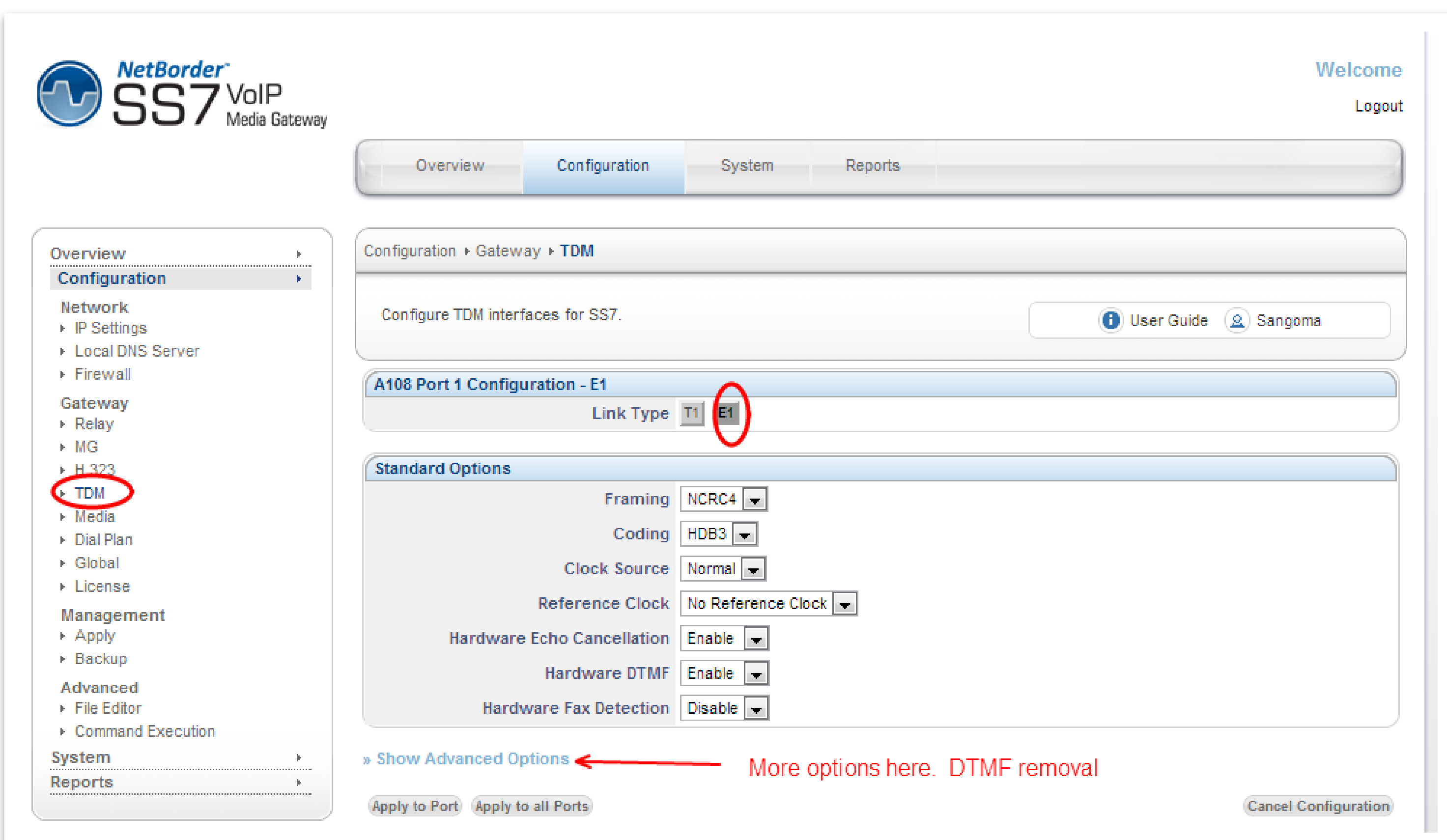
NOTE
Identify picture of the device is always set to A108D – 8 T1/E1 card. The LED will always bling port 1. The image is not meant to reflect the real hardware image, nor real port location.

Edit T1/E1 Config

- Once the port has been identified and plugged into the T1/E1 network.
- Select **Edit** button for Port 1 Card 1 to configure the physical T1/E1 parameters.
- Select the port configuration type: T1 or E1
    - T1: North American Market and Japan
    - E1: Europe and the world
- Fill in Physical Configuration T1 or E1 parameters
    - Fill in the T1/E1 parameters based on the provider provision document.

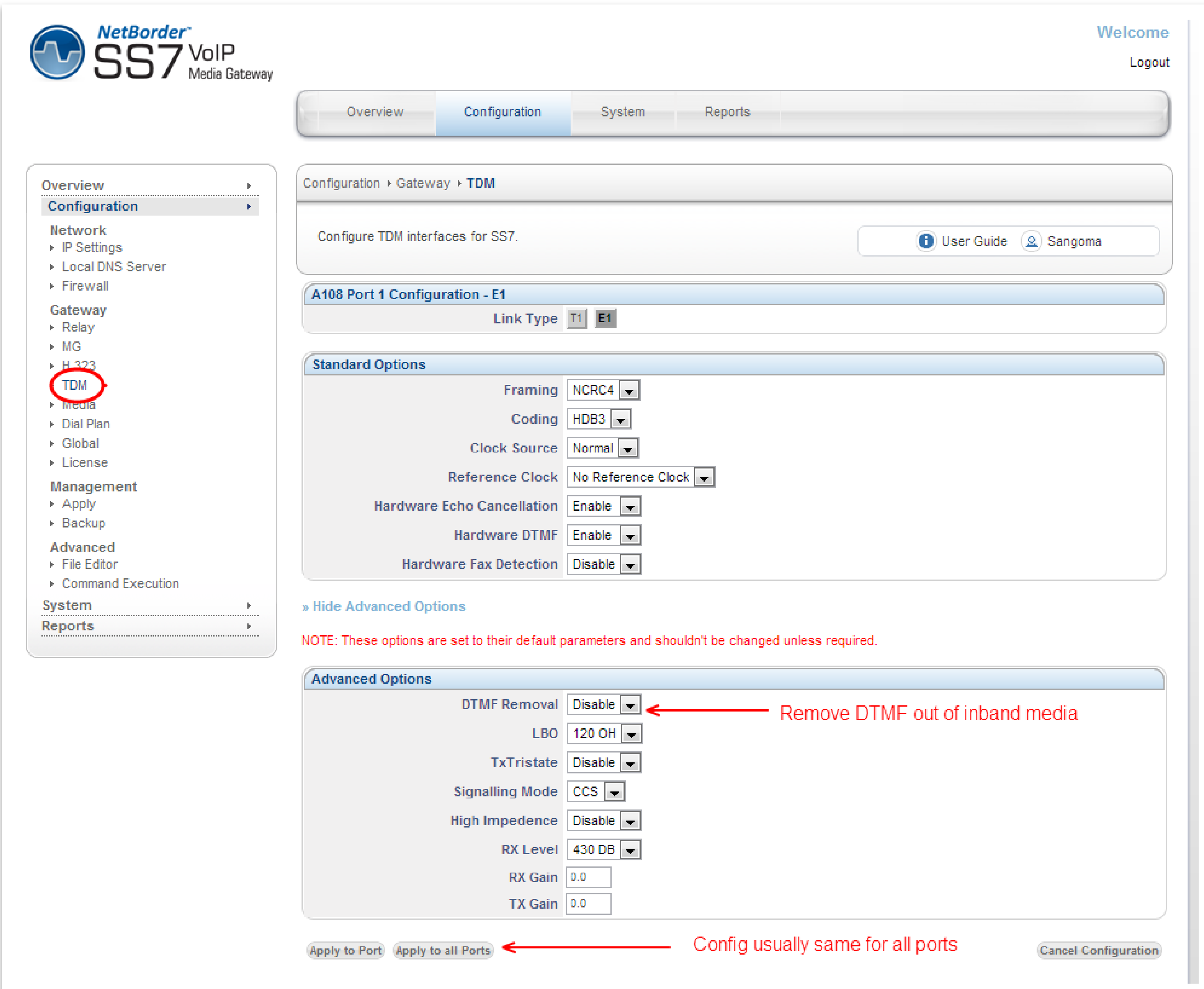


Standard T1/E1 Parameters



- In case advanced parameters are not necessary proceed
- Apply to Port
  - Applies the configuration for a single T1/E1 port
  - (The one that is currently being edited)
- Apply to all Ports
  - Apply to all T1/E1 ports on a board.
  - Bulk config feature
  - (This feature saves time as T1/E1 ports are usually provisioned the same)

Advanced T1/E1 Parameters



NOTE
After T1/E1 configuration, the NSG wizard will request **Link Type** Configuration.

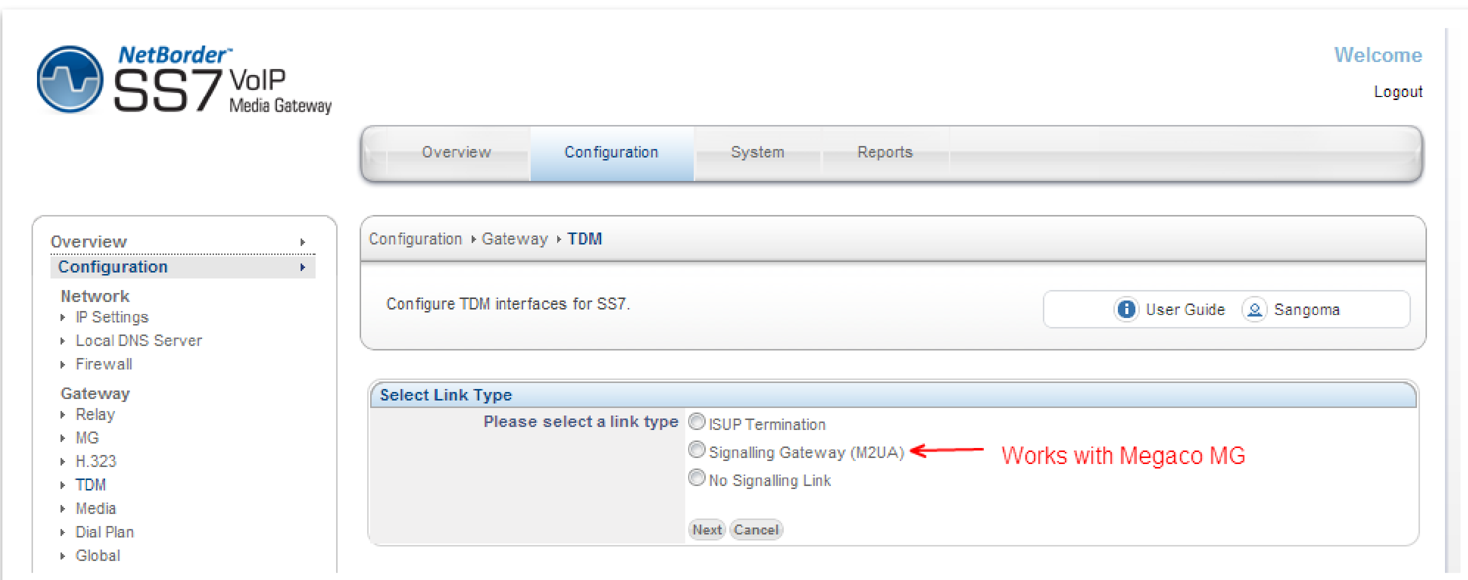
Span Link Type

When configuring TDM Terminations for Megaco Media Gateway there are two possibilities

- Voice Mode
  - All TDM channels are used for Voice 64kbs G711
  - Example: All channels 1-31 on an E1 line are used for voice
  - Link Type = Voice Only

- Mix Mode
  - Voice 64kbs G711 channels and SS7 signaling channels.
  - Example: Channel 16 is used for SS7 signaling, 1-15,17-31 are used for voice.
  - Link Type = Signaling Gateway (M2UA)

- If configuring for **Voice Mode** select **No Signaling Link**
- If configuring for **Mixed Mode** select **Signaling Gateway (M2UA)**



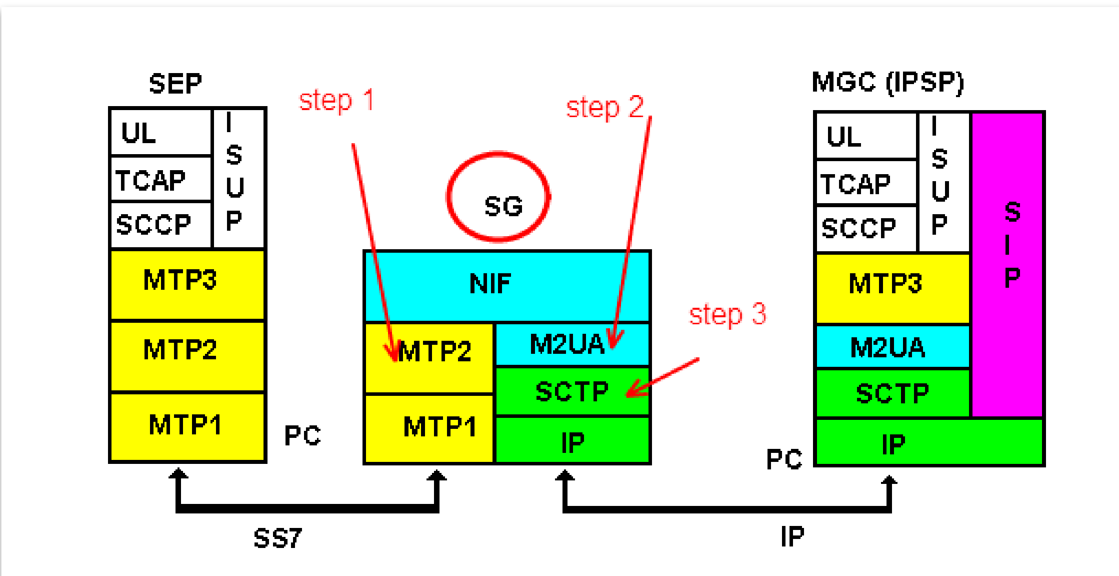
NOTE
The rest of this section will continue to document the **Signaling Gateway (M2UA)** option.
Next page will introduce the Signaling Gateway Overview, followed by the next config section in the WebGUI.

Signaling Gateway Overview

NSG supports Signaling Gateway operation mode.

In Signaling gateway mode, NSG will bridge T1/E1 SS7 signaling link to IP and pass it transparently to the MGC/Softswitch, via M2UA protocol. Looking at the diagram below, NSG Signaling Gateway will configure:

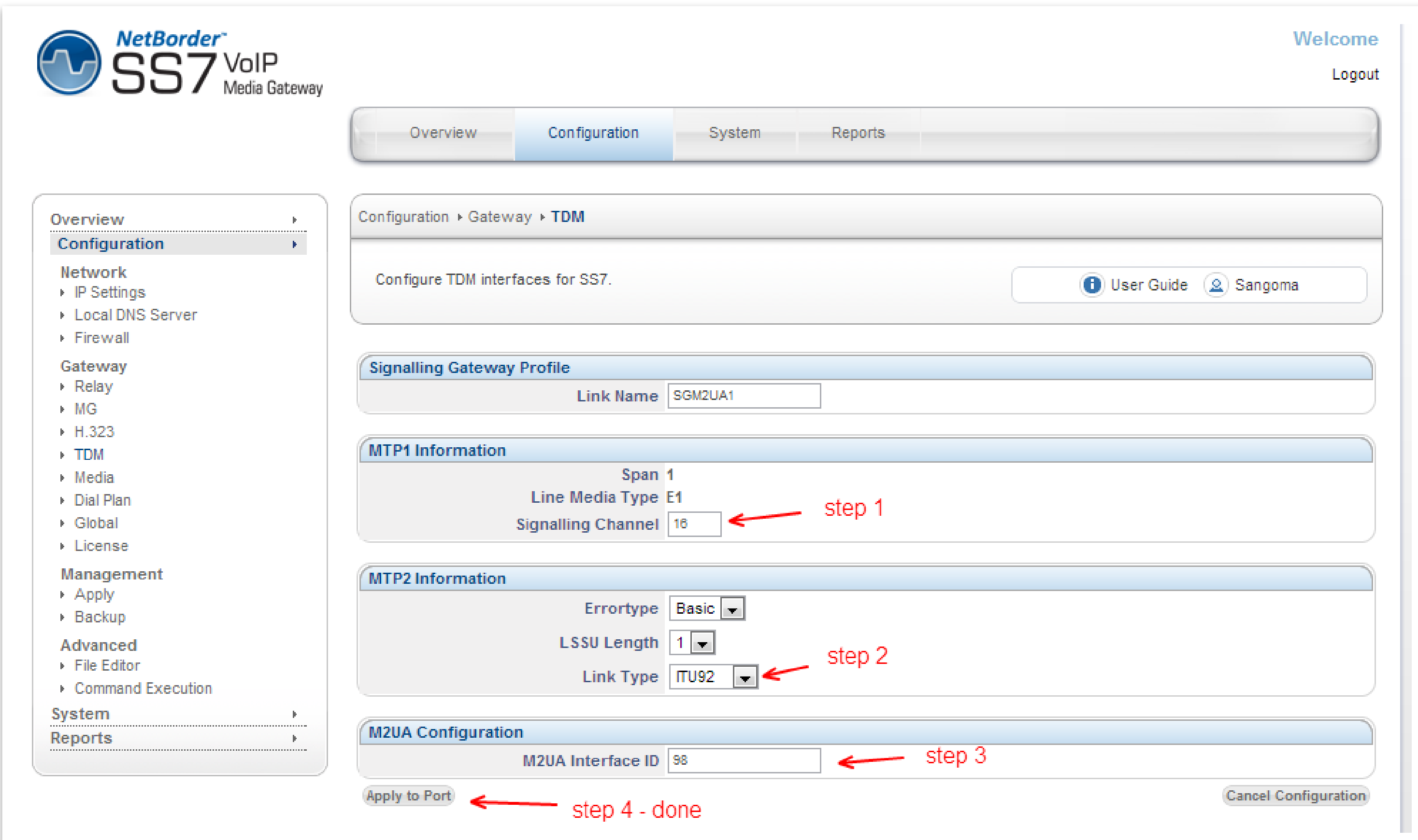
- MTP1 & MTP2 protocols over the TDM port
- M2UA/SCTP protocol over IP network
- NIF (Network interworking function) to bridge the two



100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

MTP1/2 Link Configuration

- Specify MTP1/2 information based on provider provision document
- Step1: Identify which channel on T1/E1 line will carry signaling
- Step2: Specify MTP2 signaling information based on provision document
- Step3: Specify M2UA Interface ID based on provision document
- **Apply to Port** to save configuration



100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

Field Name	*Possible Values*	*Default Value*	*Description*
Link Name	NA	NA	M2UA Profile name
Span	NA	NA	Span number which is going to associated with this M2UA profile.
Line Media Type	E1/T1	E1	Media type
Signaling channel	NA	NA	Signaling channel of the span which will carry the M2UA signaling messages.
ErrorType	Basic/PCR	Basic	MTP2 error type.
LSSU length	1/2	1	LSSU length
Link Type	ITU92 ITU88 ANSI96 ANSI92 ANSI88 ETSI	ITU92	SS7 link variant.
M2UA Interface ID	NA	NA	M2UA Interface identifier which will map to this particular signaling span/channel and uniquely identify the link between M2UA SG and MGC.

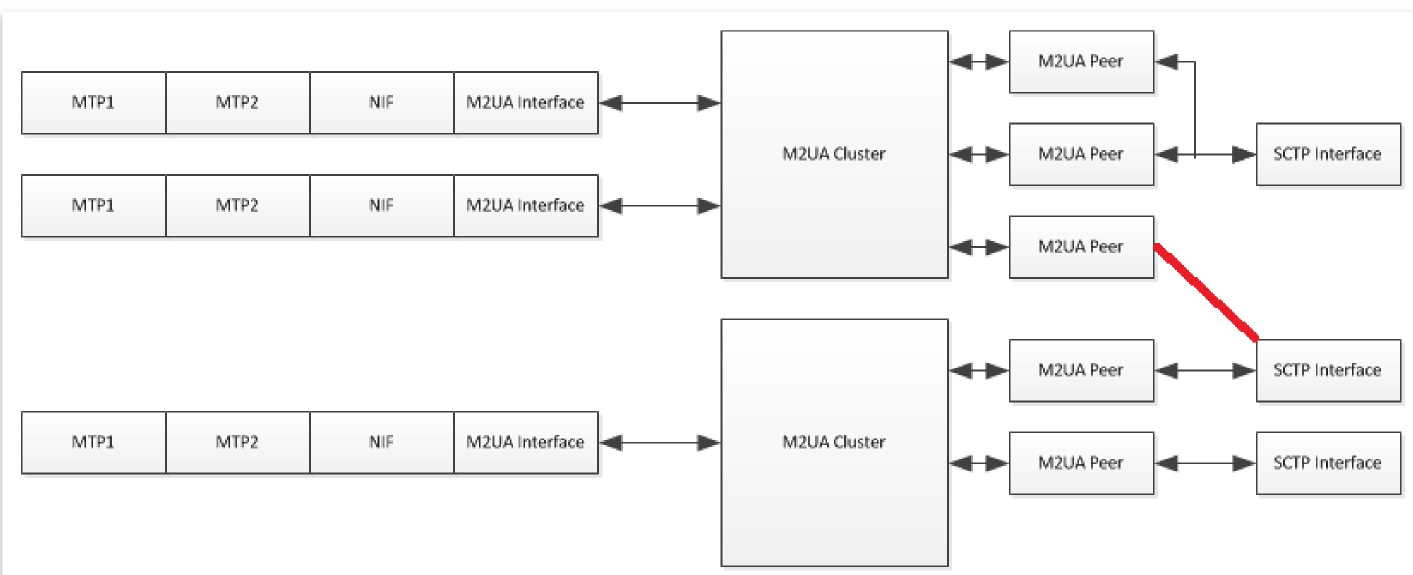
NOTE
Next section in WebUI will relate to M2UA configuration. Before we proceed however, the M2UA interface architecture will be introduced in order to provide a big picture to the user.

M2UA Interface

This section provides in-depth overview on how the M2UA interface is constructed. It should help the user better understand the WebUI configuration objects for M2UA protocol.

WebUI for M2UA contains 3 sections: Cluster, Peer and SCTP

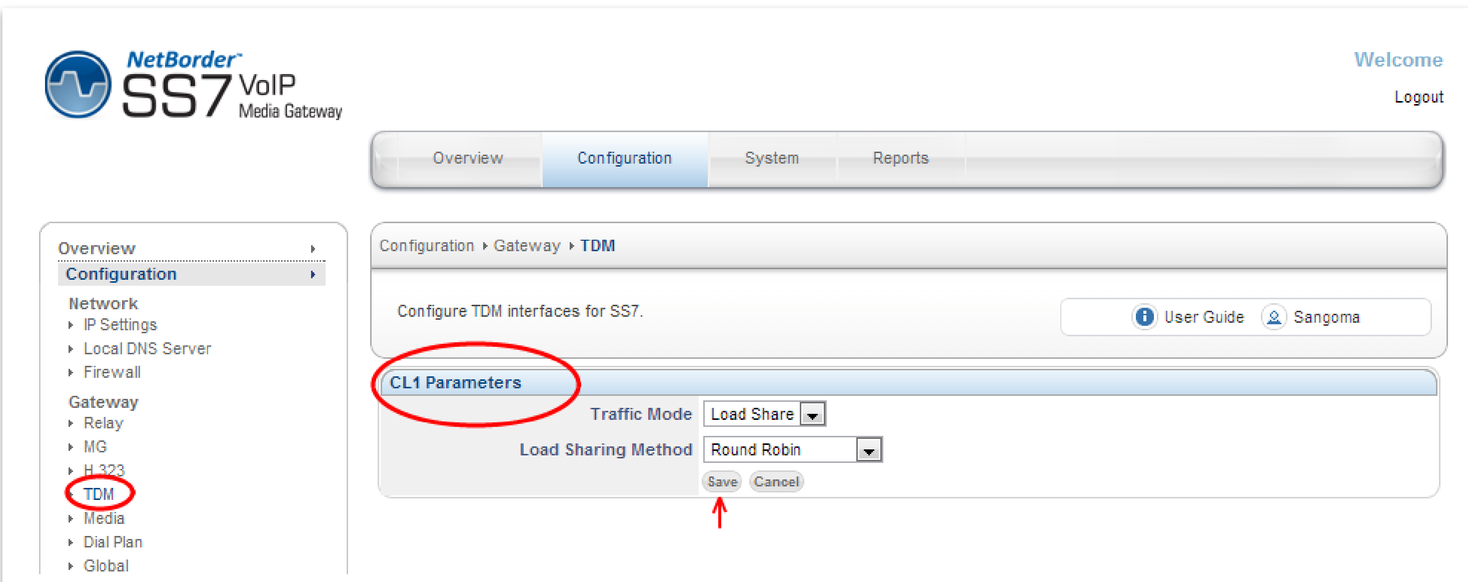
- SCTP interfaces are standalone objects on which a peer bind to (regardless of its cluster).
  - 1 SCTP binds to 1 or more peers
  - 1 peer binds to 1 SCTP
  - Thus SCTP are shared across all peers
  - SCTP cannot be deleted if used by any peer (even from another cluster).
  - Deleting a peer or a cluster does not delete SCTP.
- Peers are bound to cluster.
  - 1 peer binds to 1 cluster
  - 1 cluster binds to 1 or more peer
  - Deleting a cluster will delete peers.
- Cluster are bound to mtp2 through m2ua binding and nif interface
  - 1 cluster binds to 1 or many mtp2 (through m2ua->nif relationship)
  - 1 mtp2 binds to 1 cluster through nif interface binding



M2UA Cluster Creation

M2UA Cluster is a group of peers to which M2UA SG will communicate

- Select Create Cluster
- Leave the Cluster values default unless the provider specifies otherwise.
- Select **Save** to Continue

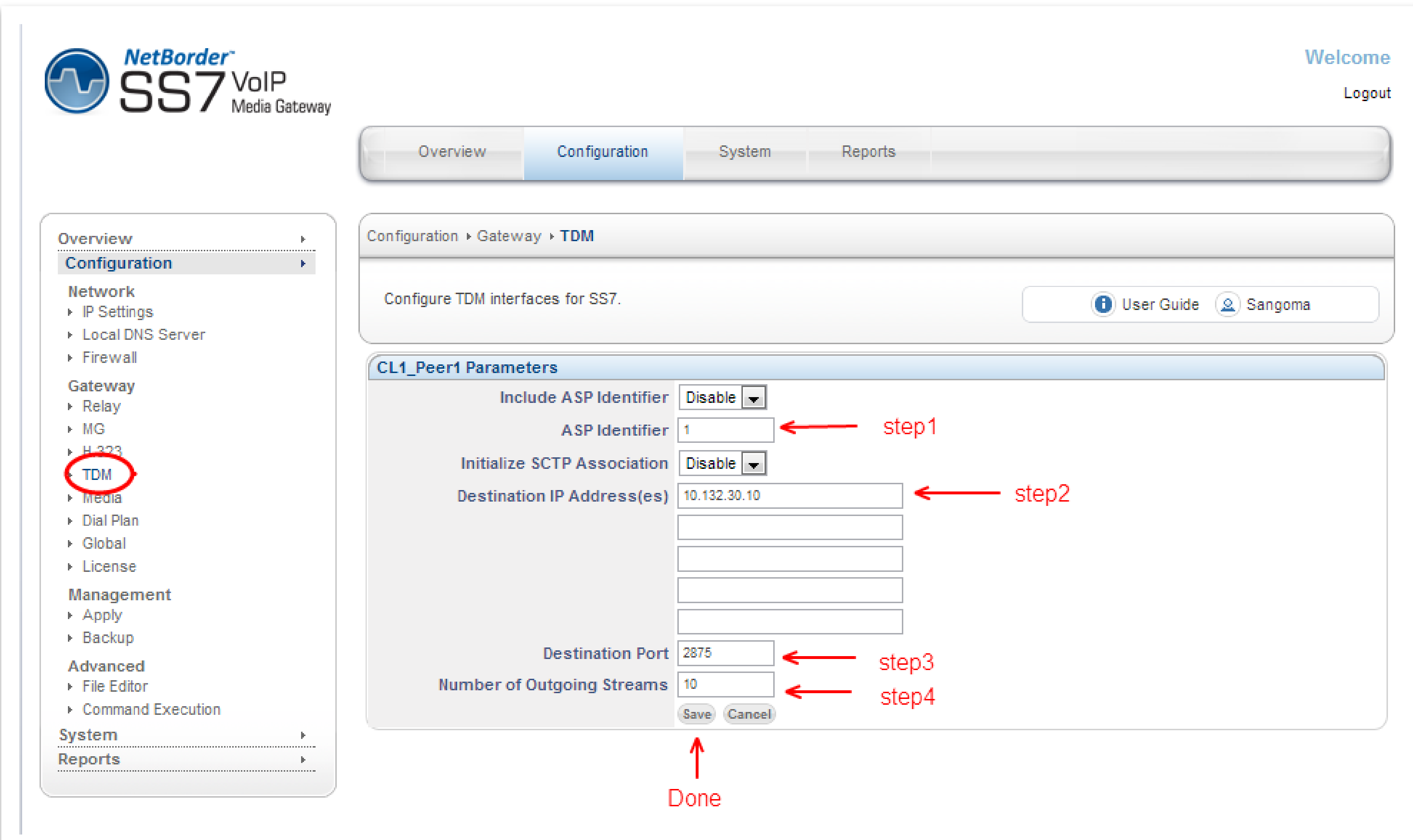


Field Name	*Possible Values*	*Default Value*	*Description*
Traffic Mode	Load Share Override Broadcast	Load Share	This parameter defines the mode in which this Cluster is supposed to work.
Load Sharing Method	Round Robin Link Specified Customer Specified	Round Robin	This parameter defines the load share algorithm which is used to distribute the traffic

M2UA Cluster Peers

M2UA Peers will be configured under the M2UA clusters

- Select **Add** under Cluster Peers Profile
- Select **Create** Cluster Peer Profile
- Specify the Cluster Peer parameters based on provider provision document

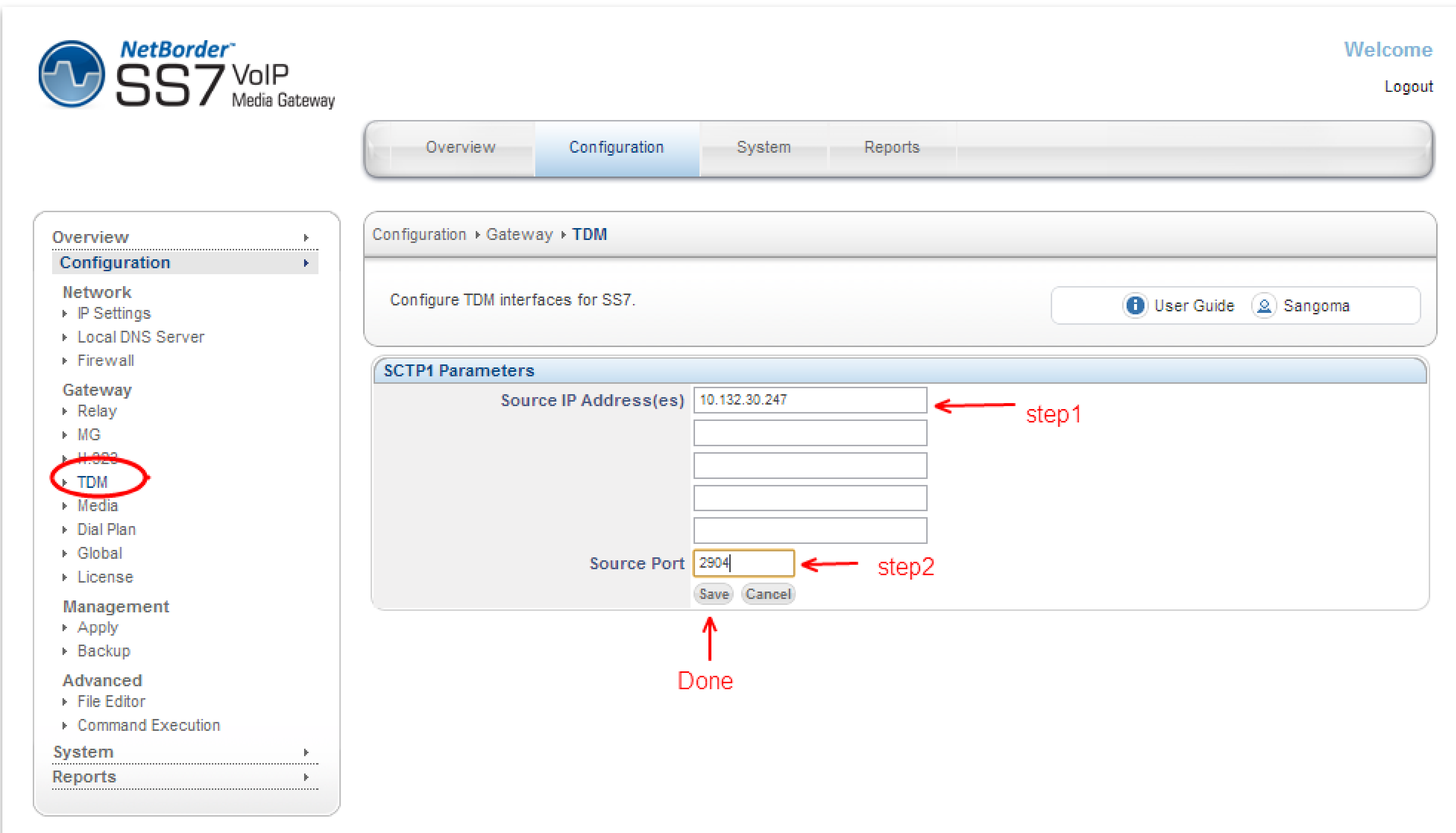


Field Name	*Possible Values*	*Default Value*	*Description*
Include ASP Identifier	Disable Enable	Disable	Flag used to indicate whether include the ASP ID in the ASP UP message
ASP Identifier	NA	NA	ASP identifier for this ASP node. Set to 1 in case ASP is Disabled
Initialize SCTP Association	Disable Enable	Disable	Flag used to indicate if M2UA SG has to start SCTP association or not. If Disable means M2UA SG will wait for SCTP association request from MGC. If Enable that means M2UA SG will initiate the SCTP association request towards MGC.
Destination IP Address(es)	NA	NA	Destination IP address
Destination port	NA	NA	Destination Port
Number of Outgoing Streams	NA	10	Number of outgoing streams supported by this association. Default 10

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

SCTP Interface

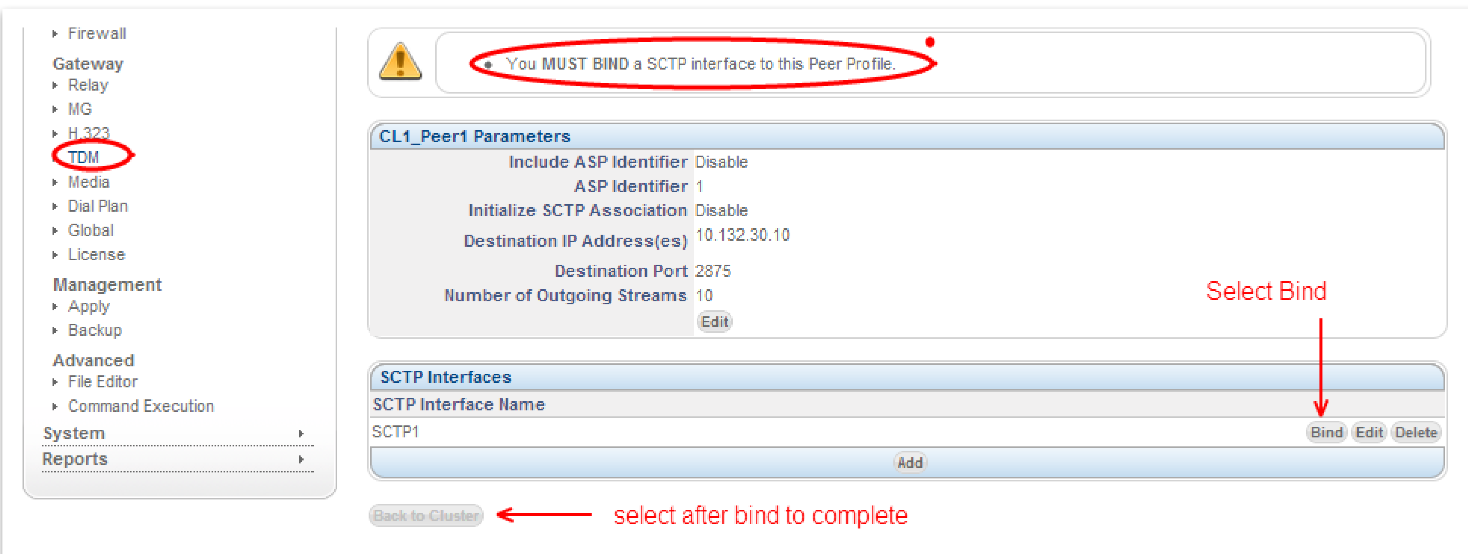
- Select Add SCTP Interface
- Select Create SCTP Interface
- Specify SCTP Information based on provider provision document

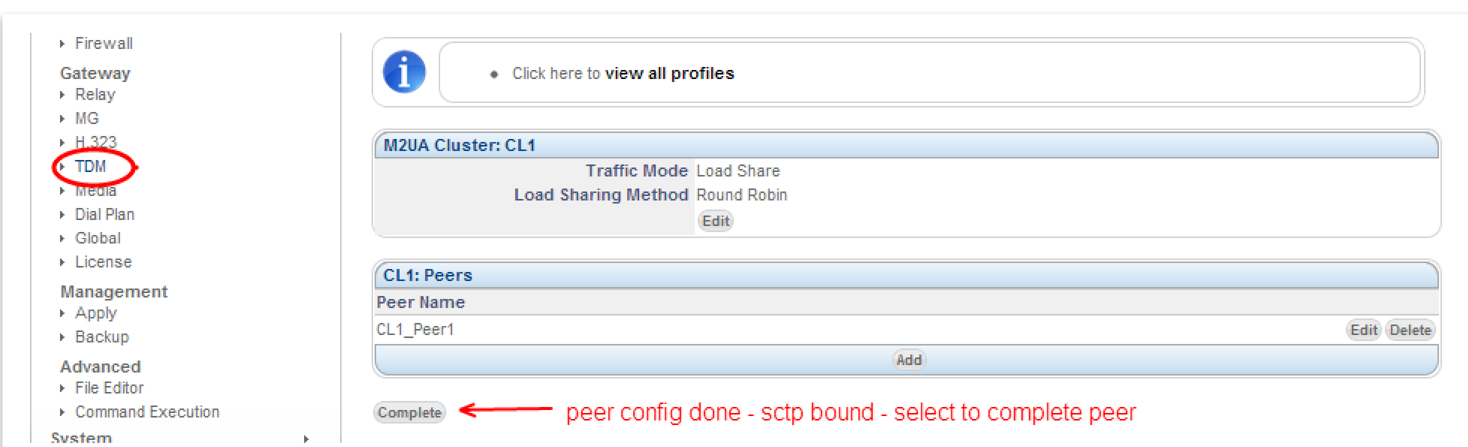


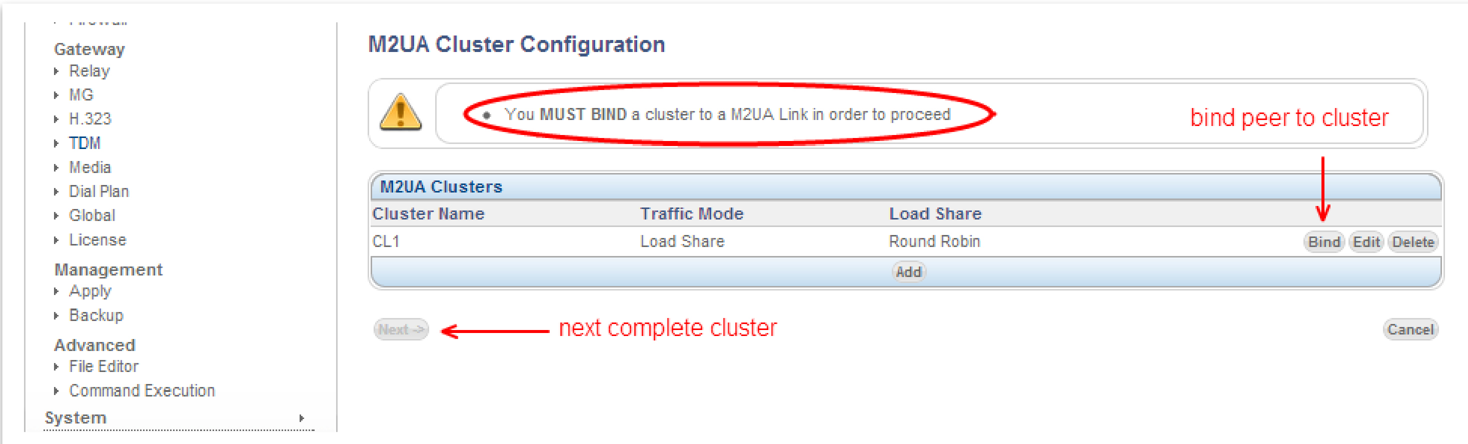
Binding all components

- All components have been created
  - M2UA Cluster
  - M2UA Peer
  - SCTP Interface

- Next step is to Bind / Connect them together
  - SCTP interface into M2UA Peer
  - M2UA peer into M2UA Cluster

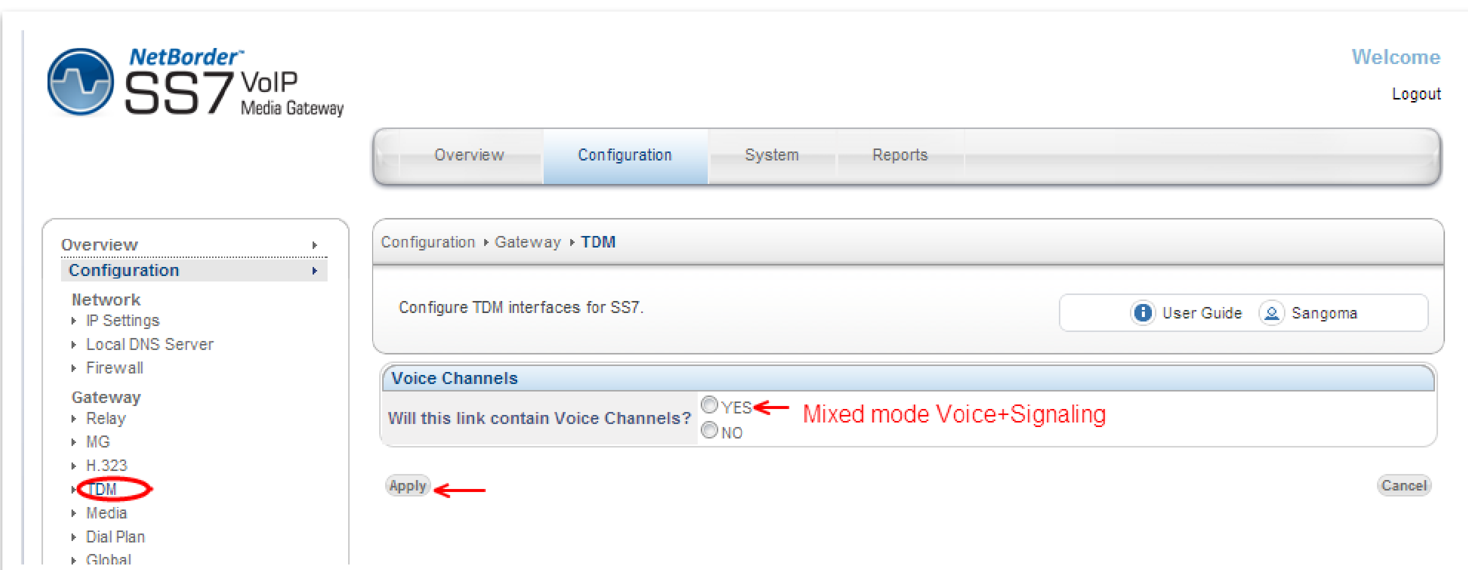






Mixed Mode Configuration

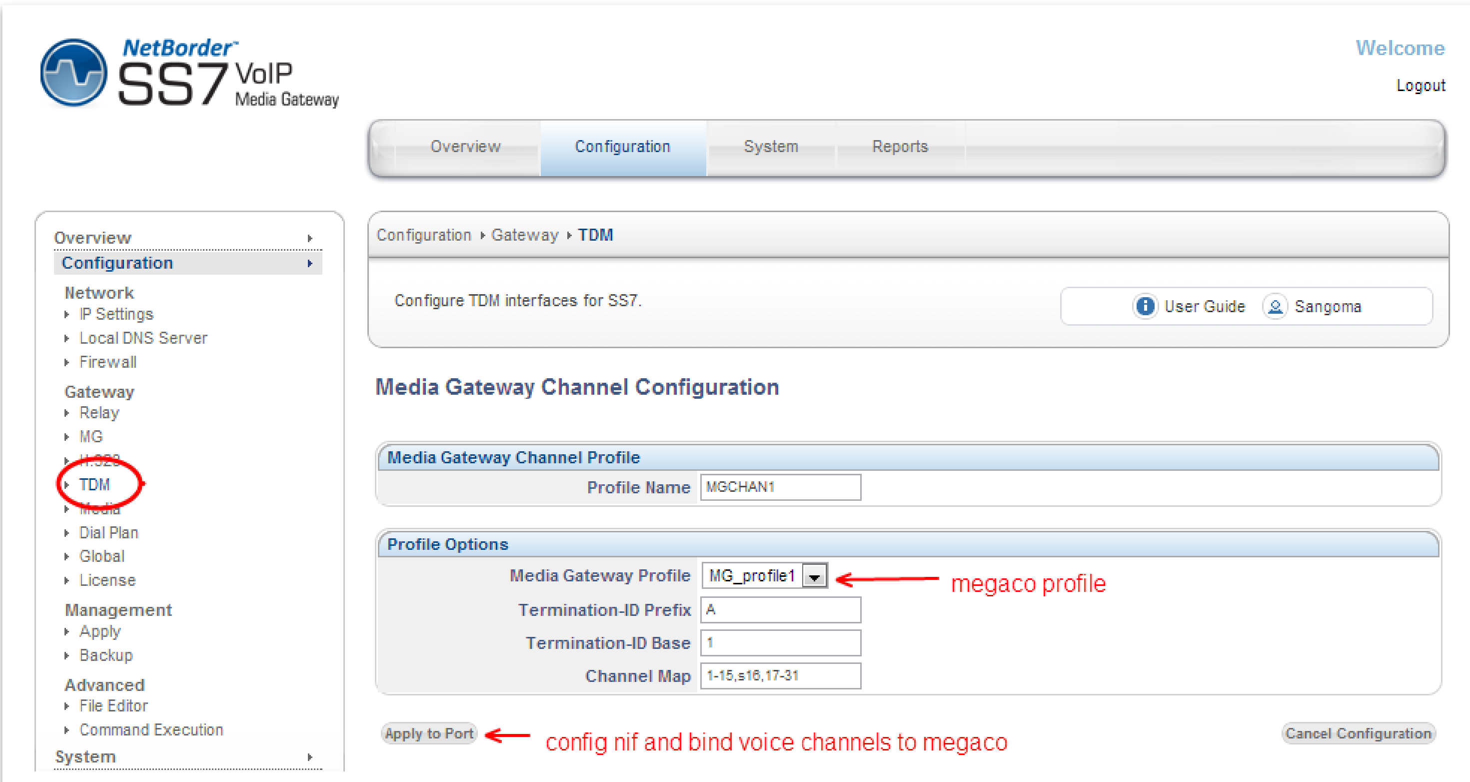
- Signaling is bridged by M2UA to the MGC/Soft switch
- Voice is controlled by Megaco/H.248
- Specify that Voice is part of this TDM Span



NOTE
Rest of this section will document the **Mixed Mode Configuration**

Bind Megaco to TDM

The last step of the configuration is to bind the TDM voice channels to Megaco Profile.

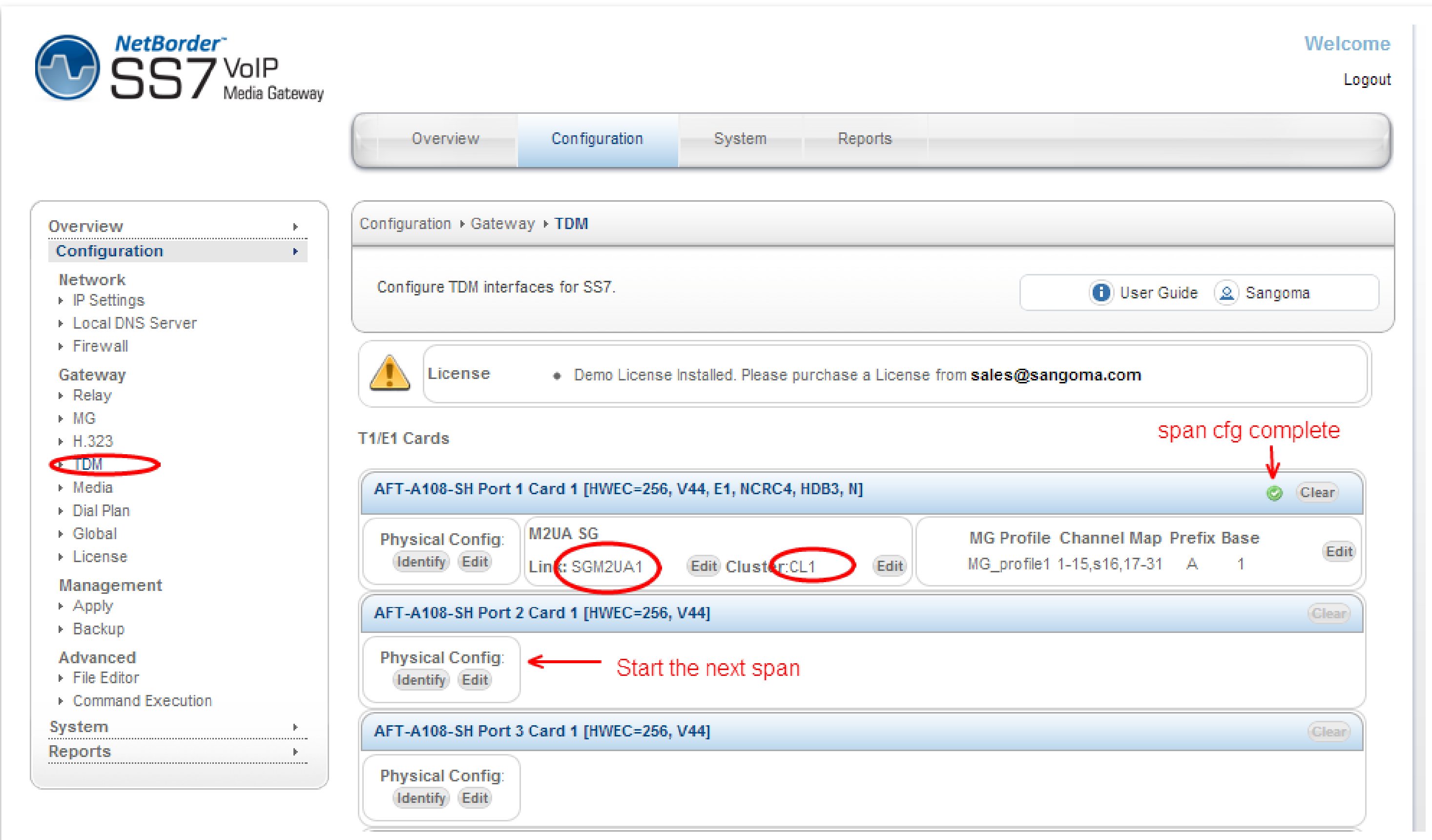


Field Name	*Possible Values*	*Default Value*	*Description*
Media Gateway Profile	List of Gateways	First in the List	Select Megaco Profile that will be used to control the TDM channels for this span.
Termination ID Prefix	NA	NA	Usually a letter A-Z. This prefix is defined by MGC. Please refer to MGC configuration.
Termination ID Base	NA	NA	Usually a number starting from 1. This value is defined by MGC. Please refer to MGC configuration.
Channel Map	NA	NA	List of channels to be controlled by Megaco Example: 1-15,s16,17-31 Channels 1-15 and 17-31 are used for Voice and should be controlled by Megaco Channel 16 (prefixed by letters) indicates that channel 16 carries signaling channel. Megaco will ignore this channel as it's not voice. The bind between megaco and tdm would be as follows A1: channel 1 A2: channel 2 A3: channel 3 … A16: not used – signaling channel A17: channel 17 A18: channel 18 … A31: channel 31

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

TDM Termination Complete

- A span has been configured and bound to a Megaco Profile.
- Configuration for this span is done
  - Confirmed in WebUI by a green checkmark.



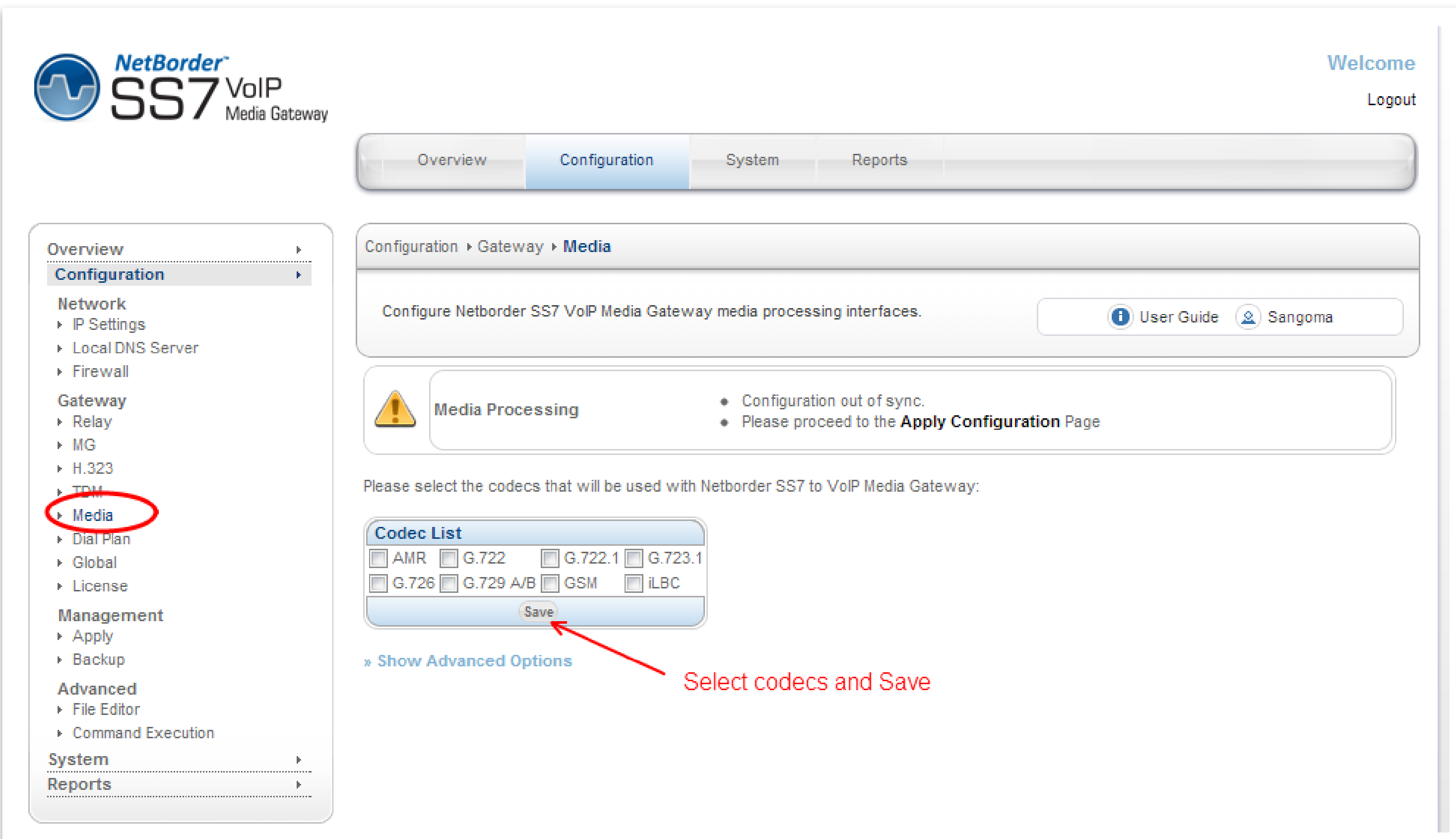
- Next step is to repeat the process for the rest of the spans.
- In typical configurations there is one or two spans (T1/E1 ports) that contain signaling channels. The rest of the spans are usually voice only.
- In voice only config, there is no Signaling Gateway configuration.
  - The configuration jumps directly to "Bind TDM to Megaco" section of the WebUI.

NOTE
The changes made in the Configuration section of the WebUI are only stored one the scratch disk. User MUST proceed to Apply page in the Management Section to save new configuration.

8. Media Transcoding Configuration

To access NSG Media Transcoding Configuration

- Select Media from side/top Configuration Menu
- Select any or all supported/listed codecs
- Once done press **Save**

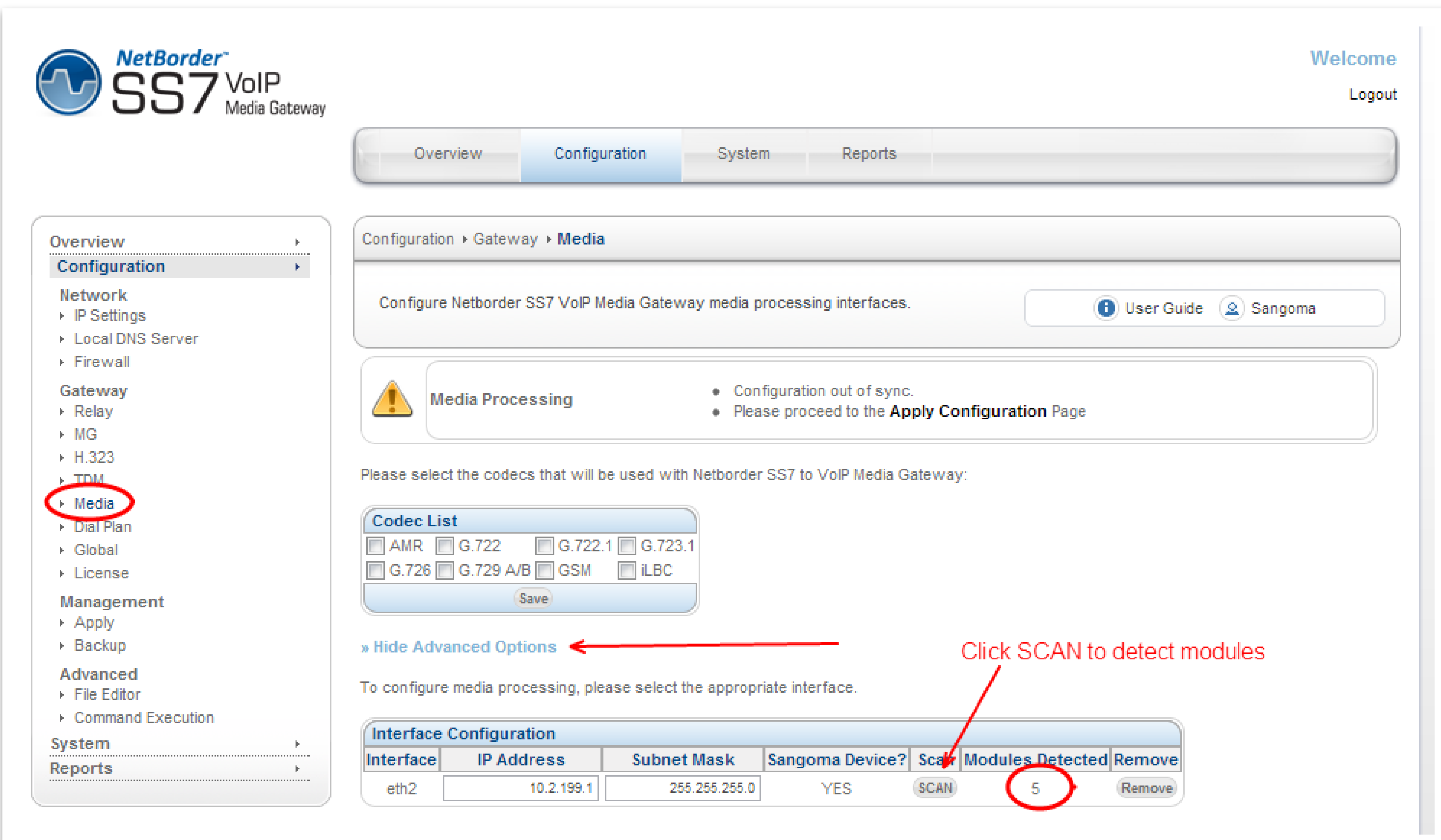


NOTE
At this point the codec selection is over. We must proceed to Media hardware discovery in the Advanced Options of the Media page.

Media Hardware

Once Codec selection has been made, proceed to Advanced Options section of the Media page.

- Select SCAN
  - This step will autodetect all NSG transcoding resources
- Confirm that GUI detected exact number of transcoding resources as installed.



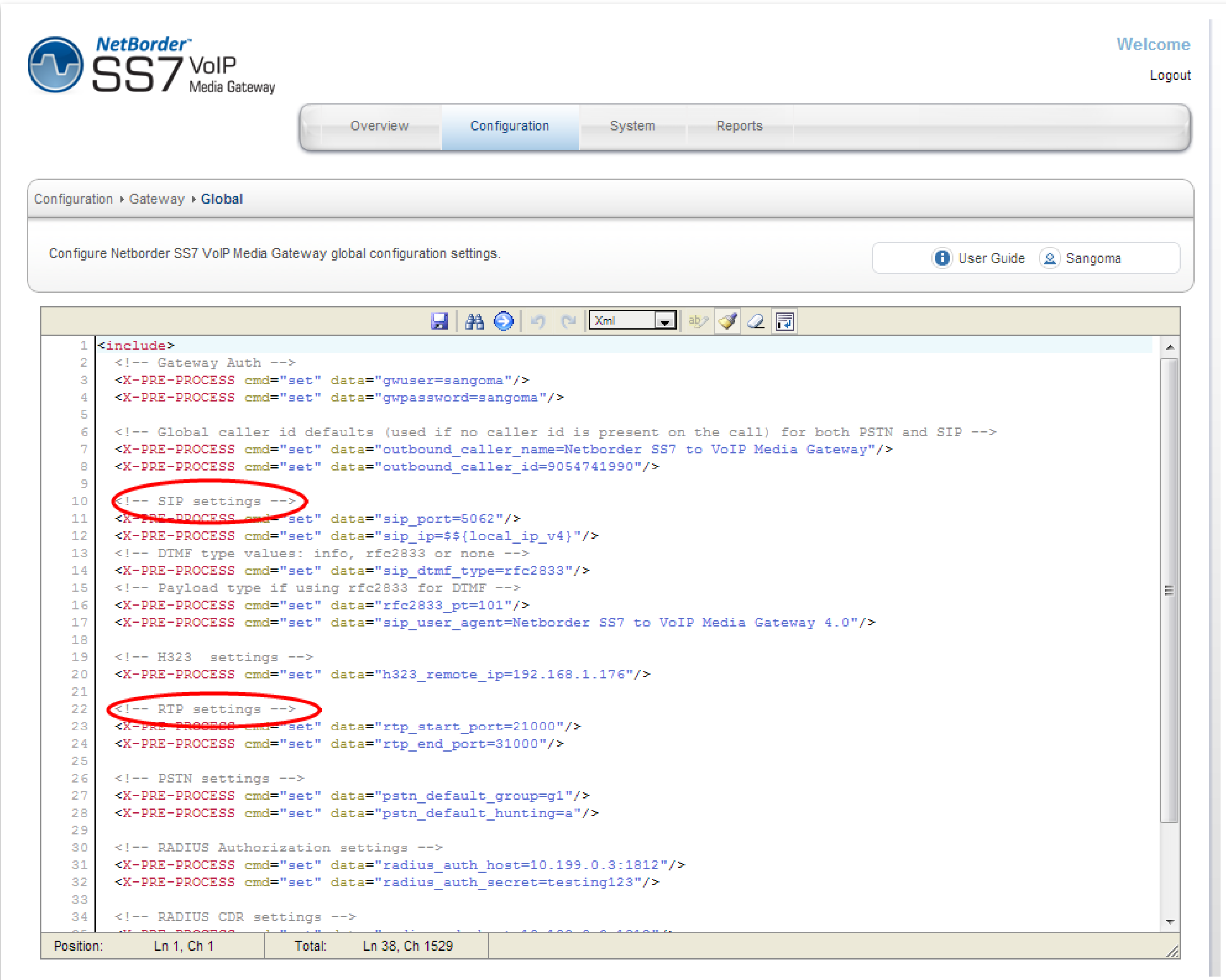
NOTE
At this point the Media configuration is complete.

- Proceed to the next section, or
- If finished all gateway configuration, proceed to Apply to generate configs.

9. SIP Endpoint Configuration

To access VoIP: SIP configuration section

- Select **Global** from side/top **Configuration** Menu
- Change a SIP global variable and Click on Save (Disk Icon)
- Proceed to Control Panel and Restart the VoIP Gateway.

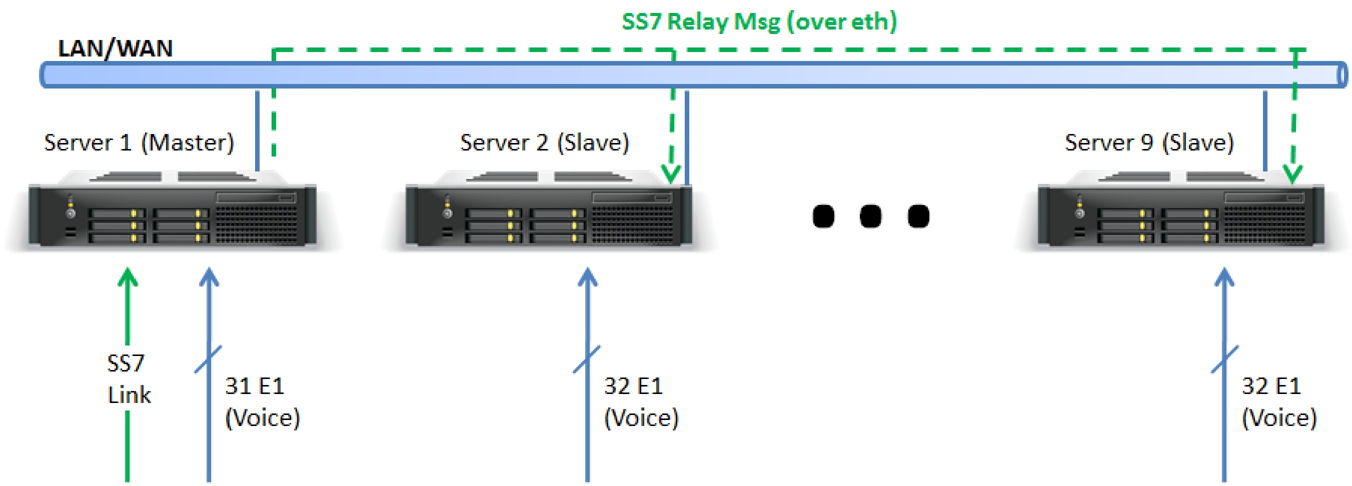


Field Name	*Possible Values*	*Default Value*	*Description*
gwuser	Any string	sangoma	NSG SIP incoming registration authentication user name.
gwpassword	Any string	sangoma	NSG SIP incoming registration authentication password
outbound_caller_name	Any string	Netborder SS7 to VoIP Media Gateway	Global caller id name defaults (used if no caller id name is present on the call) for both PSTN and SIP
outbound_caller_id	Any digits	9054741990	Global caller id defaults (used if no caller id is present on the call) for both PSTN and SIP
sip_port	Any port number	5062	SIP service port number.
sip_ip	Any ip address	System IP	SIP service IP address. By default a system eth0 address is taken as default ip address.
sip_dtmf_type	rfc2833 info none	rfc2833	rfc2833 - DTMF passed via RTP oob message info - DTMF passed via SIP INFO message none - DTMF passed via inband media
rfc2833_pt	Any number	101	rfc2833 rtp payload type override. Ability to set the RTP payload type for rfc2833. Use d edge cases where remote equipment is not per spec.
sip_user_agent	Any string	Netborder SS7 to VoIP Media Gateway 4.0	SIP INVITE user agent name string.
rtp_start_port	Any port	21000	RTP port starting range value. NSG will pick RTP ports for each call within this range.
rtp_end_port	Any port	31000	RTP port stop range value. NSG will pick RTP ports for each call within this range
pstn_default_group	g1,g2,g3,g4 ….	g1	Default pstn dial group number, in case the group is not specified in the dial string.
radius_auth_host	Any ip address:port	10.199.0.3:1812	Location of the Radius server, that will be used to authenticate incoming calls.
radius_auth_secret	Any string	testing123	Password of the remote Radius server.
radius_cdr_host	Any ip address:port	10.199.0.3:1812	Location of the Radius server, that will be used to keep track of billing via CDRs.
radius_auth_secret	Any string	testing123	Password of the remote Radius server.

10. Relay: SS7

NSG SS7 relay enables a single NSG server (master) to control multiple NSG servers (slaves) with as little as 1 signaling link connected to the master.

You can have up to 8 slave machines that are controlled by a single master server. Signaling messages (MTP2 traffic) are passed over the IP network to the slave machines.



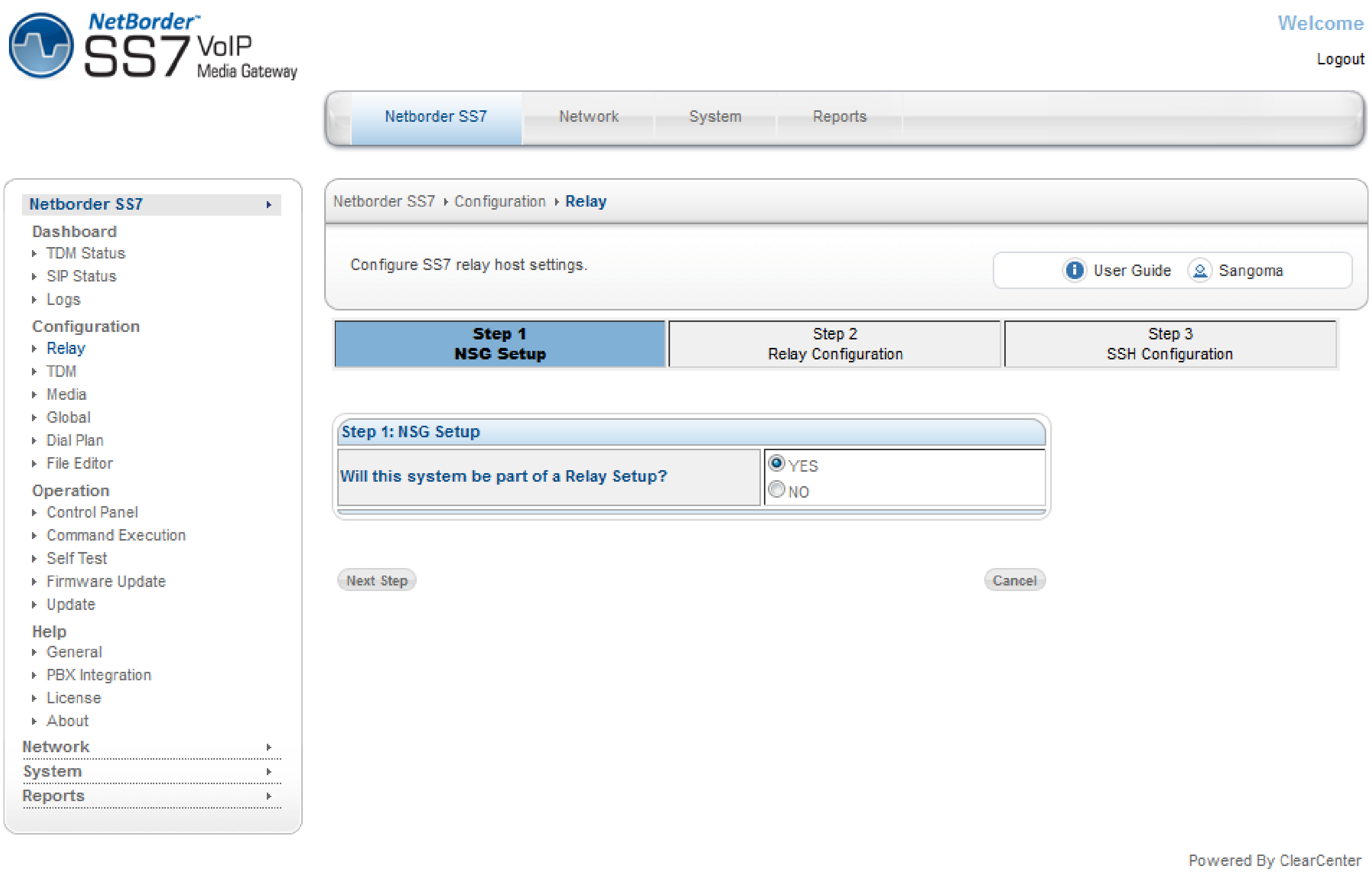
Having to configure up to 8 machines individually would be a tedious task from an operations perspective. In order to simplify the configuration process of this distributed system, the relay option enables the Master server to configure all the slaves machine from its web UI and pushing the configurations to the slave servers over SSH.

This following section will guide you through the configuration of the Relay mode to enable remote control of the Slave servers.

Relay Configuration

To access the Relay: SS7 configuration section

1. Select **Relay** from side/top **Configuration** Menu

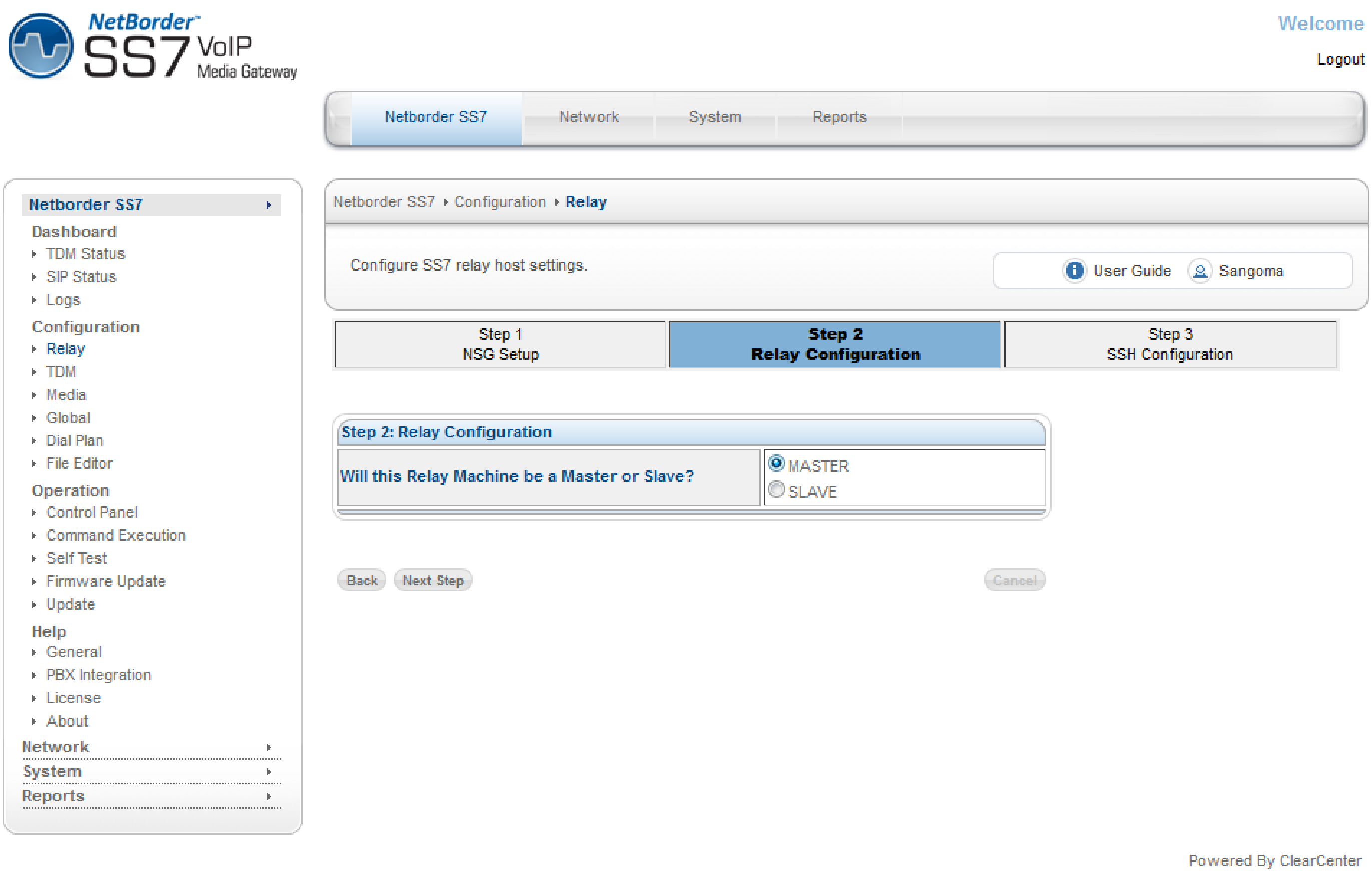


- Select **No** if you do not want to enable Relay mode in your installation and proceed to the next section to resume SS7 configuration.

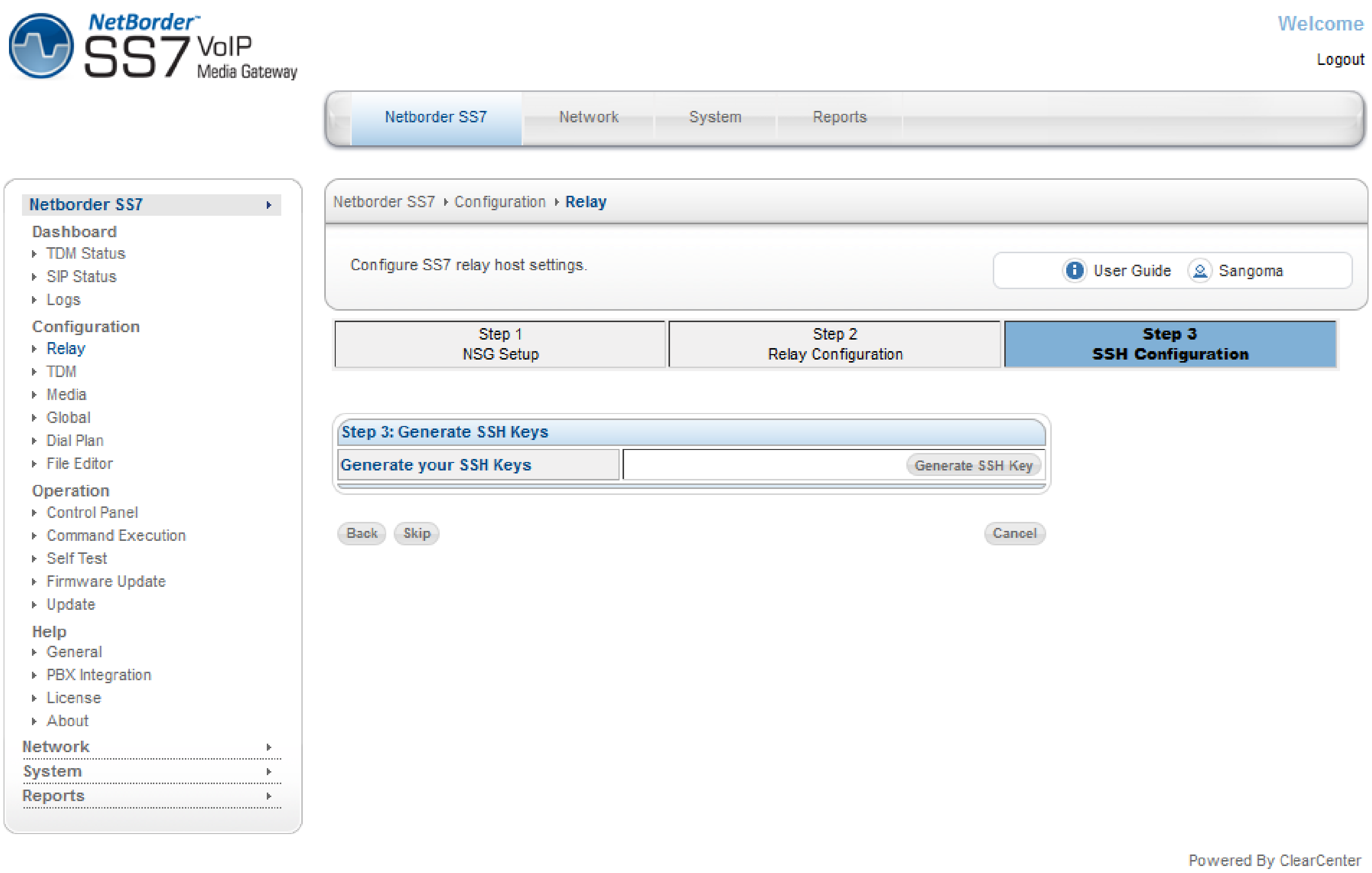
- Select **Yes** to activate the relay Mode

Configuring the master server

We will start by configuring the master machine first.

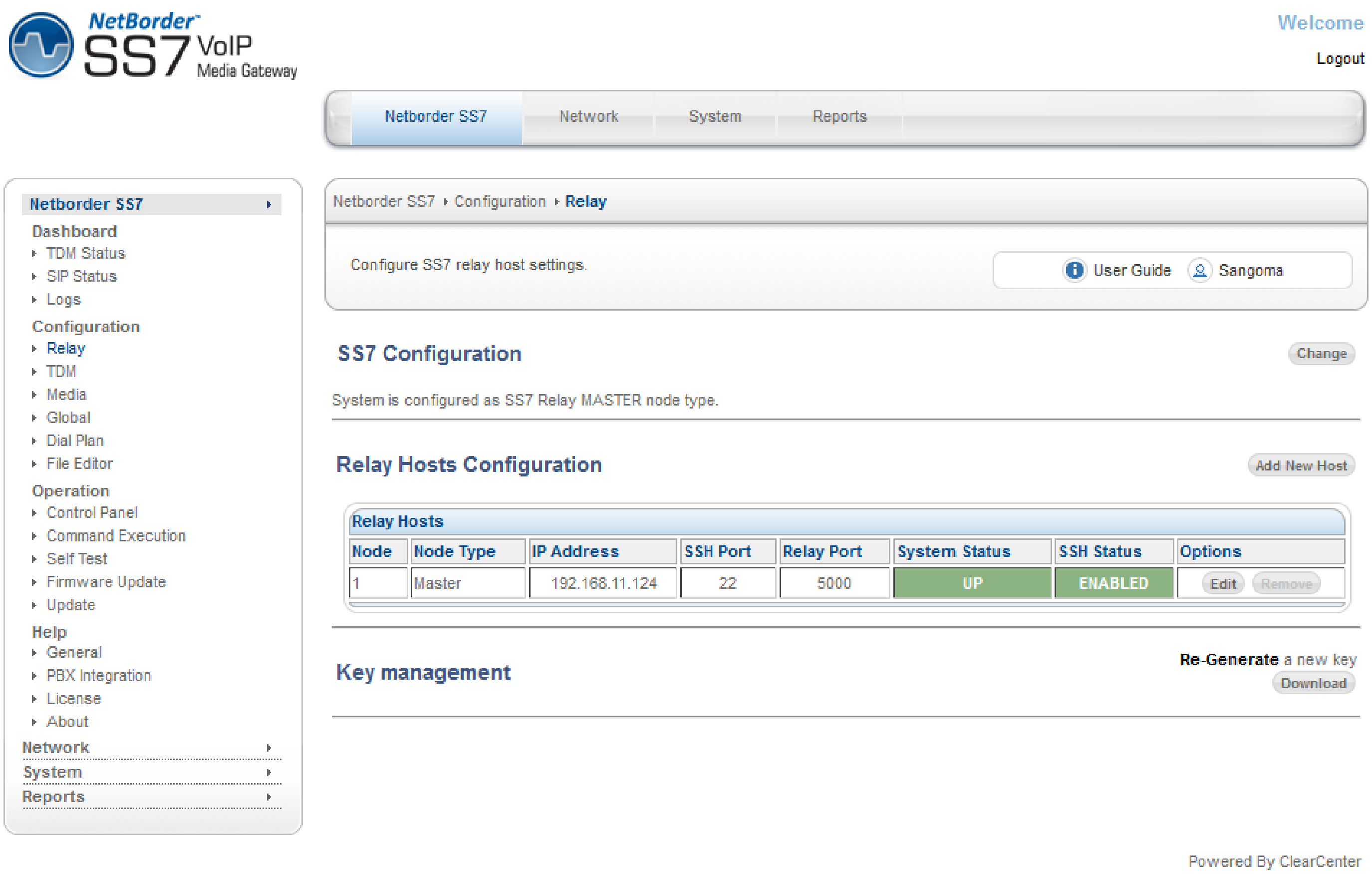


Select the Master option in step 2 and click "Next Step" to continue.



In Step 3, you will generate an SSH key and download the public key that will be uploaded to all the slave servers. This key will enable a secure SSH connection between the master and the slave machines to push the configurations.

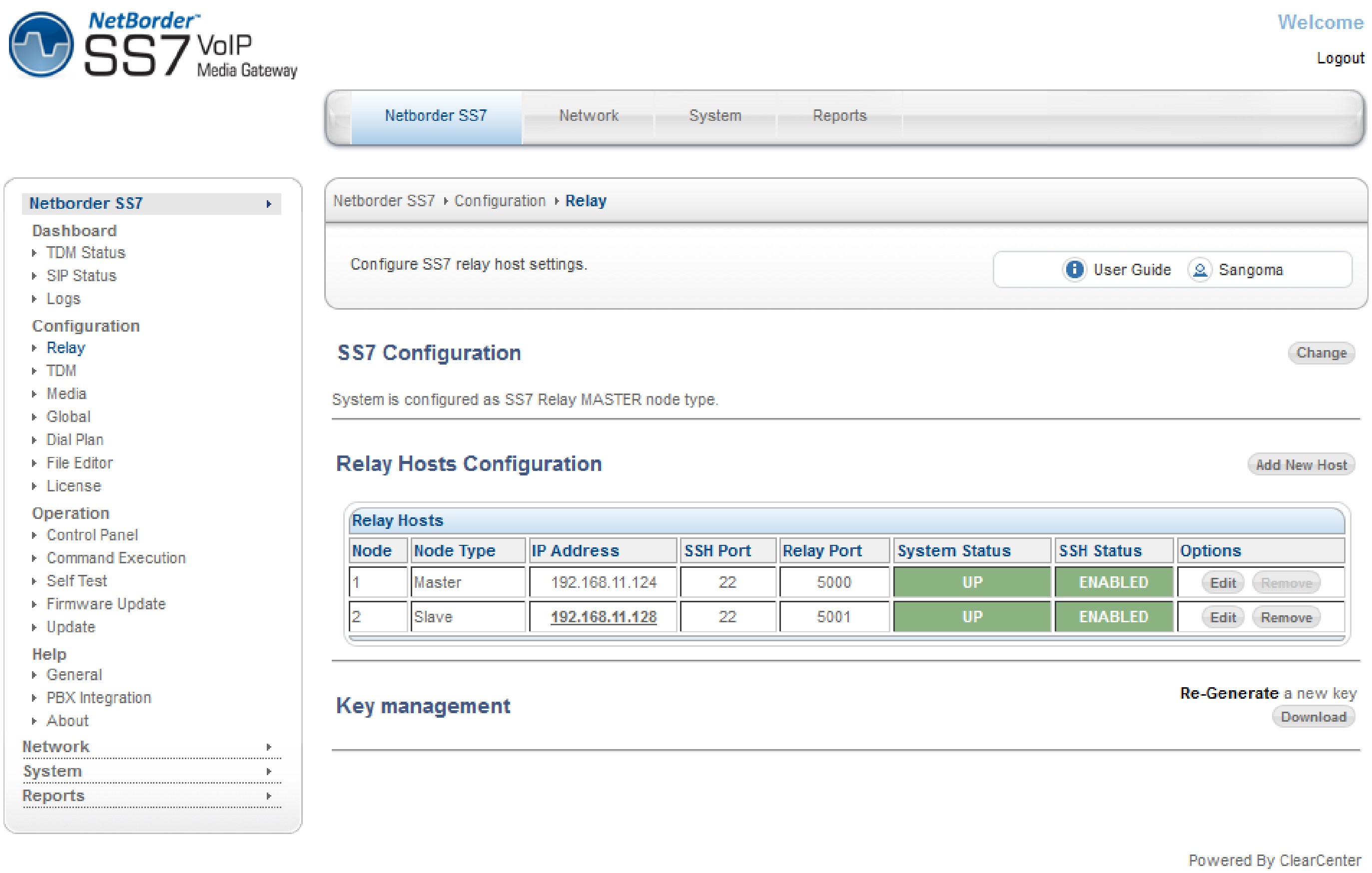
The Relay Master will listen for incoming relay traffic on port 5000.



Once the SSH key has been generated you will need to click on the "Add New Host" button to add 1 or more slave servers to the relay configuration.

The listening relay port for all subsequent slave instances will increase by 1 port. Slave on node 2 will listen on port 5001, Slave on node 3 will listen on port 5002, etc...

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

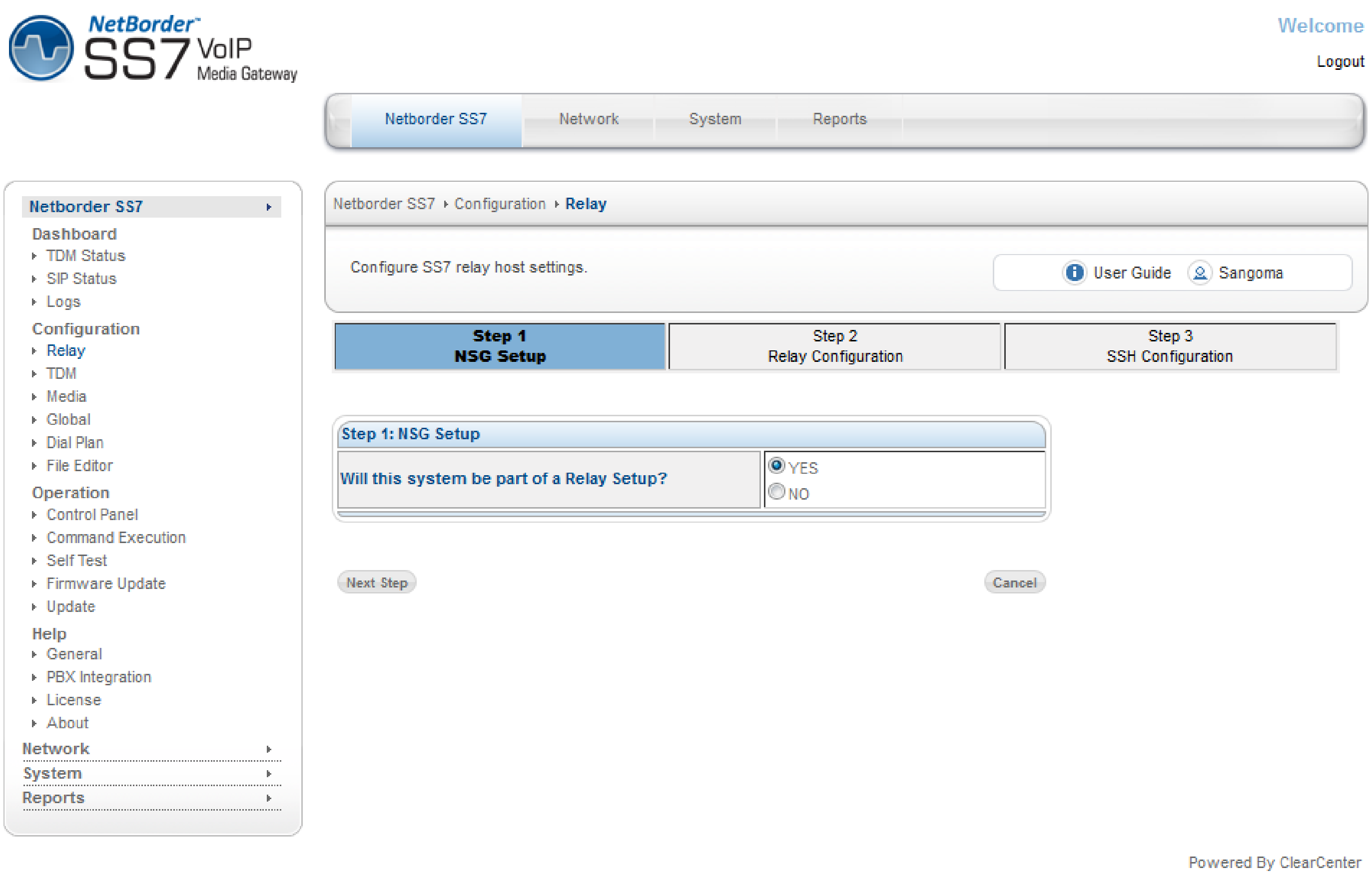


Once you have configured all your slave hosts, you can now configure your slave machine(s)

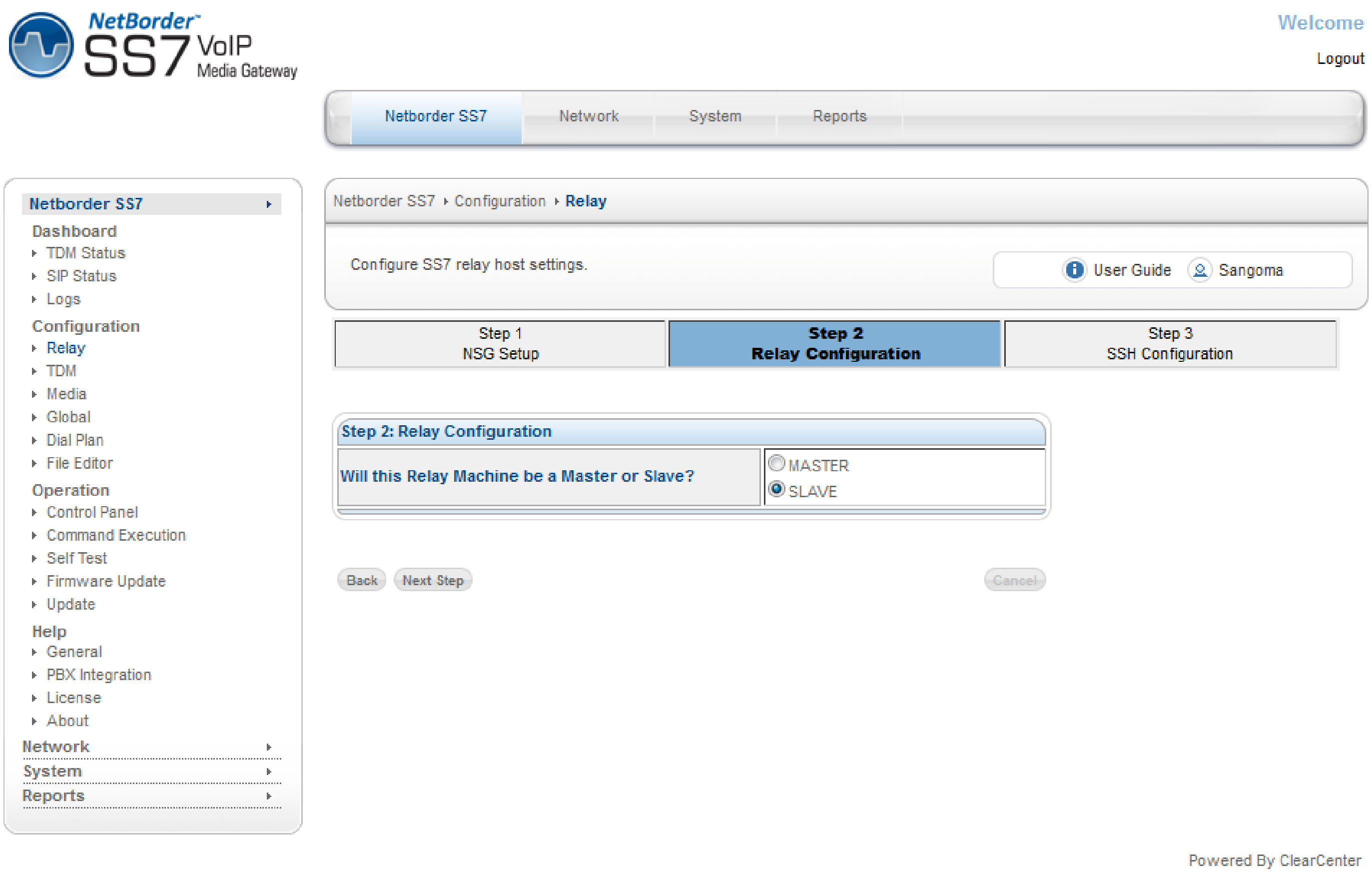
Configuring the slave server

To access the Relay: SS7 configuration section

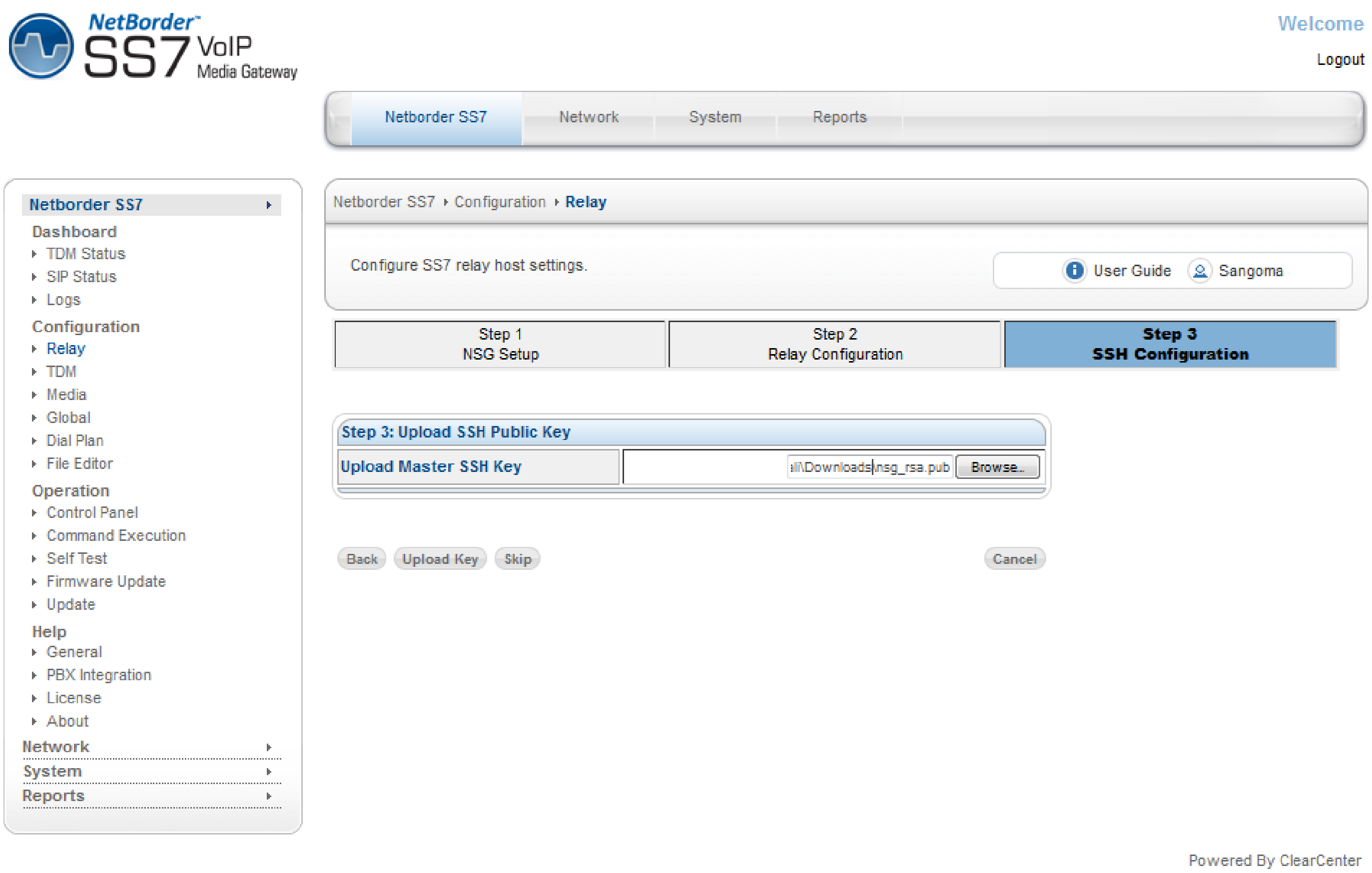
1. Select **Relay** from side/top **Configuration** Menu



Select Yes in step 1 to enable Relay mode.

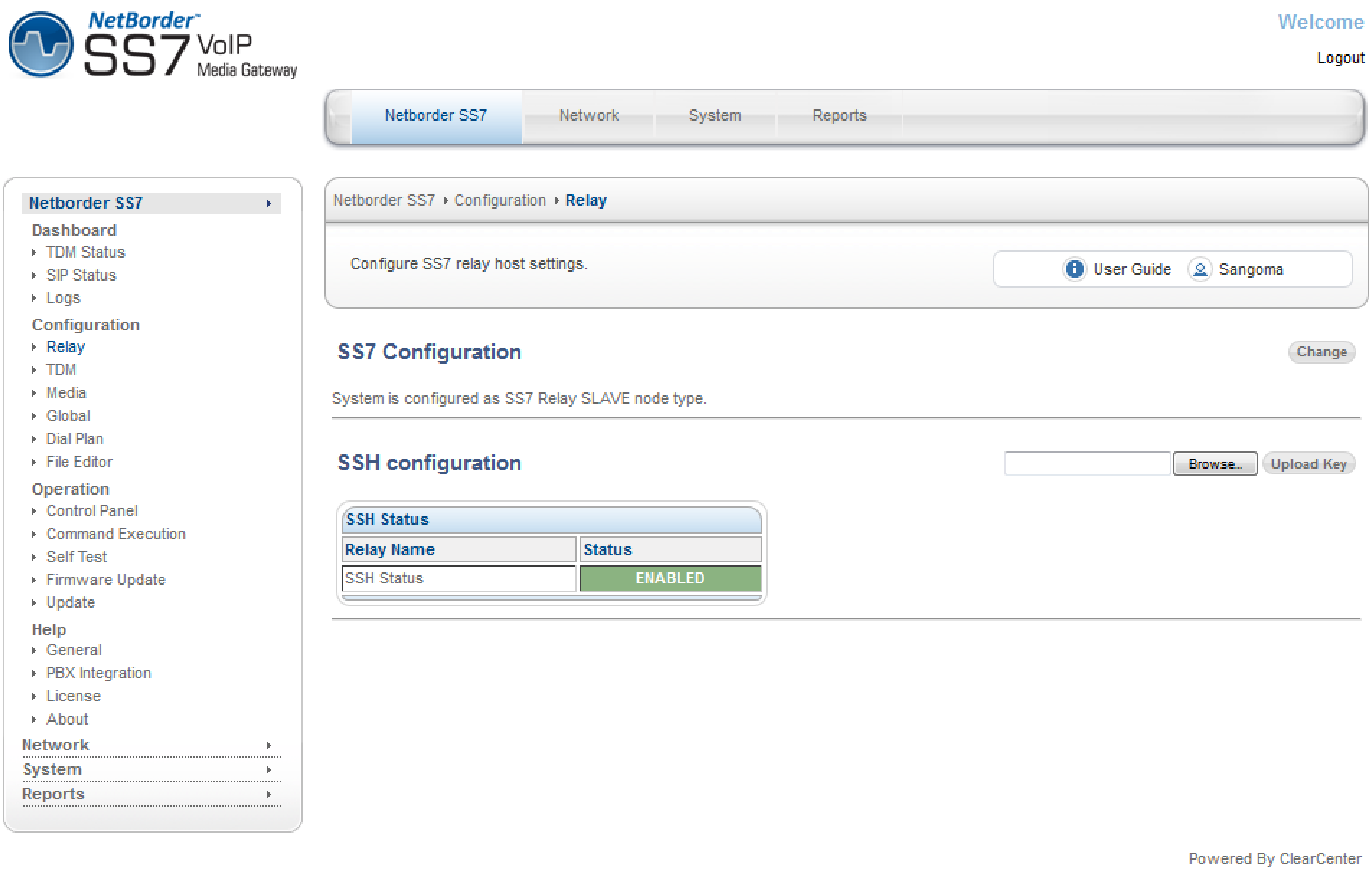


Select the Slave option in step 2 and click "Next Step" to continue.



Upload the public key that you downloaded and saved when you configured the master server earlier.

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com



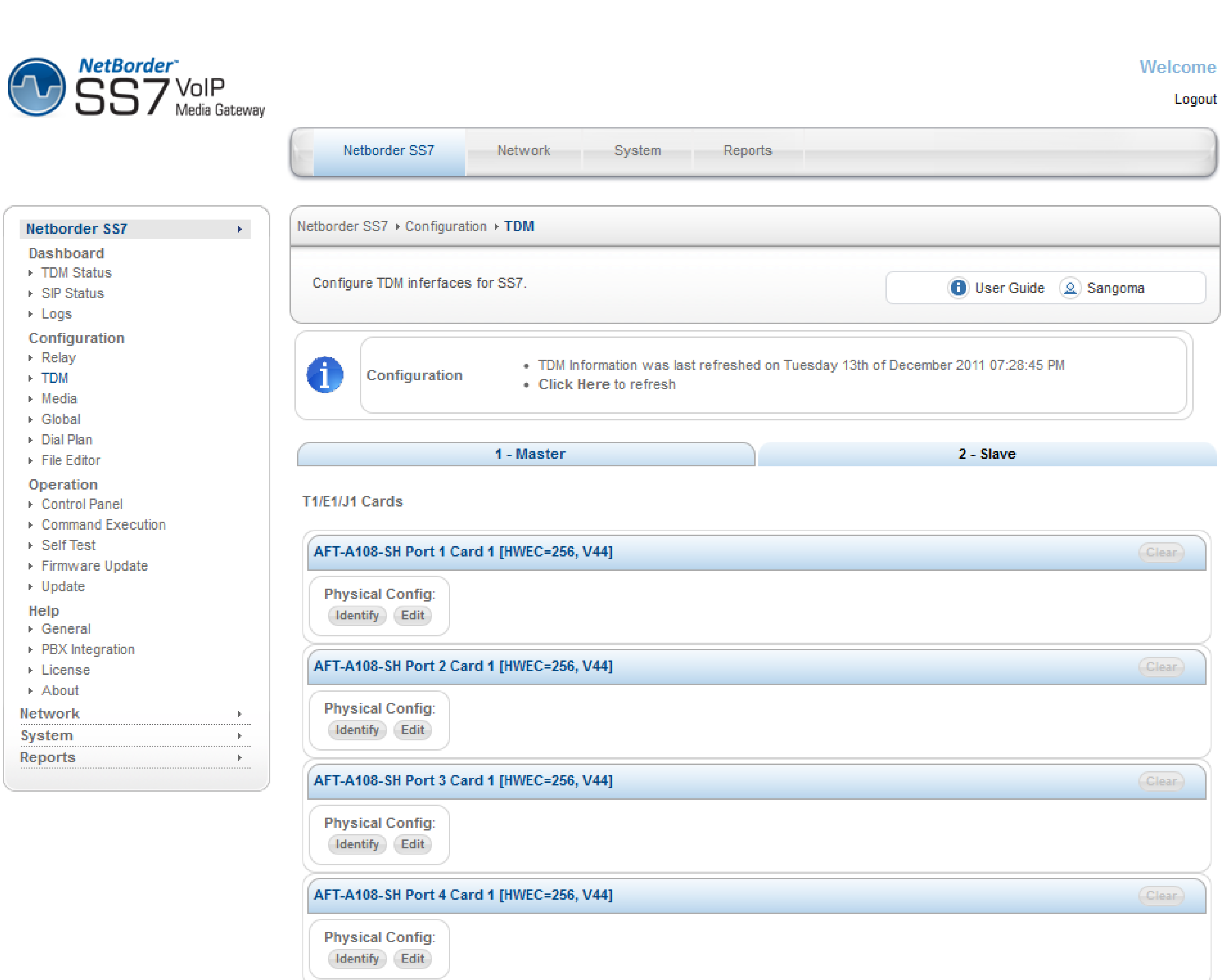
Once the key has been uploaded, the SSH link will have been enabled.

Repeat these steps for all the slave machines and return to the master web gui when you are finished

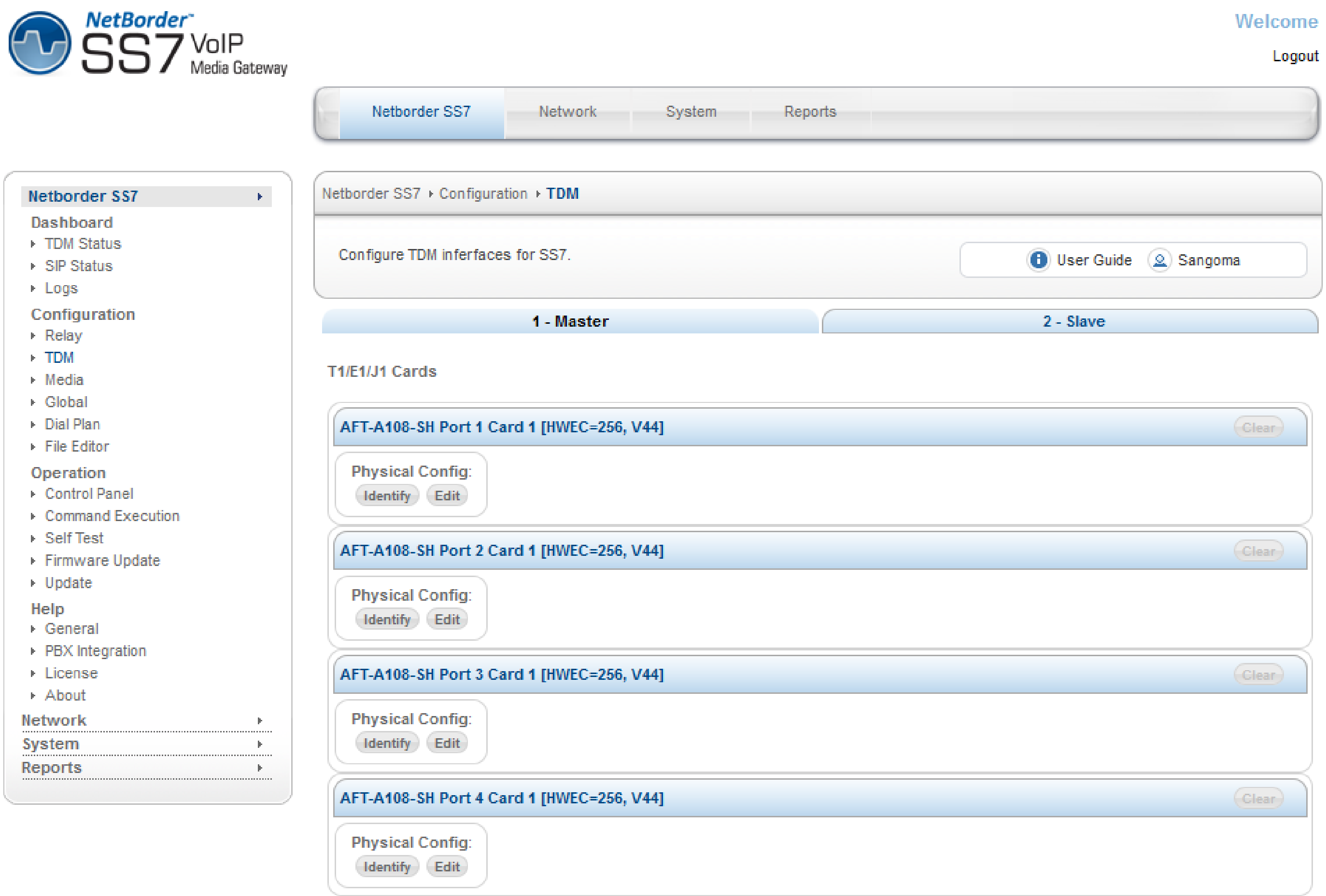
Configuring the slave TDM configurations from the master server

Open the master web gui in your browser.

1. Select **TDM** from side/top Configuration Menu



The TDM configuration is presented in a tabbed pane, each tab represents a machine to configure. Select the slave tab to configure the slave server.

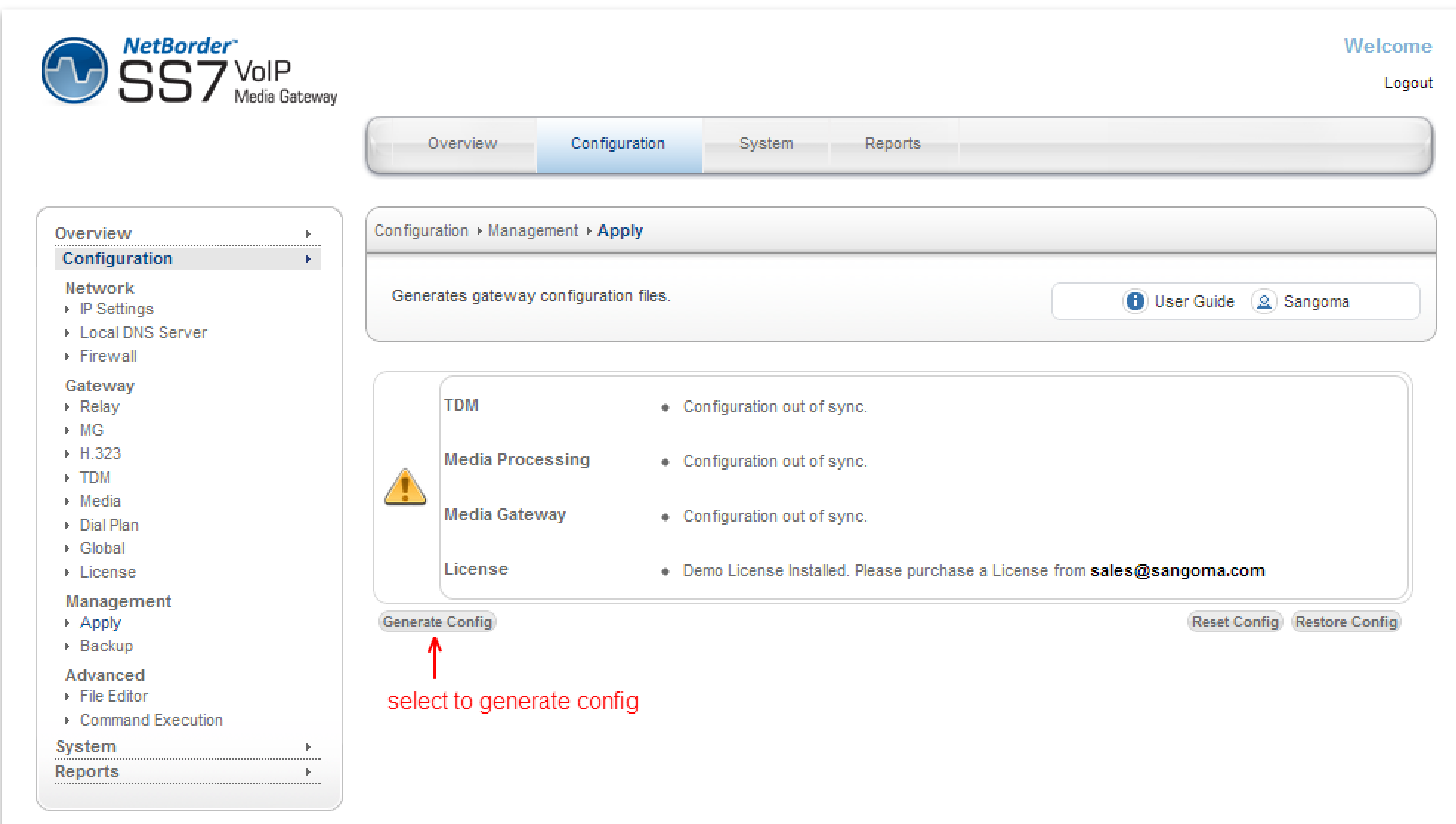


Once you have completed configuring the master and slave(s) TDM configurations, you will click on the "Generate config" button that will push the configuration to each slave over a secure SSH connection. All this is done from the convenience of the master server's web gui, removing the need to log on to each slave server's web gui individually.

11. Applying Configuration

The changes made in the **Configuration** section of the WebUI are only stored one the scratch disk. User MUST proceed to Apply page in the Management Section to save new configuration.

- Select **Apply** from side/top **Configuration** Menu
- Visually confirm the warnings
  - License warning need to be resolved with Sales
- Select **Generate Config** to apply the configuration to file/disk.
  - Generate Config will generate all necessary NSG SS7 VoIP Gateway configuration files needed to successful start the NSG gateway.



NOTE:
After configuring the NSG endpoint/protocol configuration, proceed to **Dialplan** to configure the routing rules.

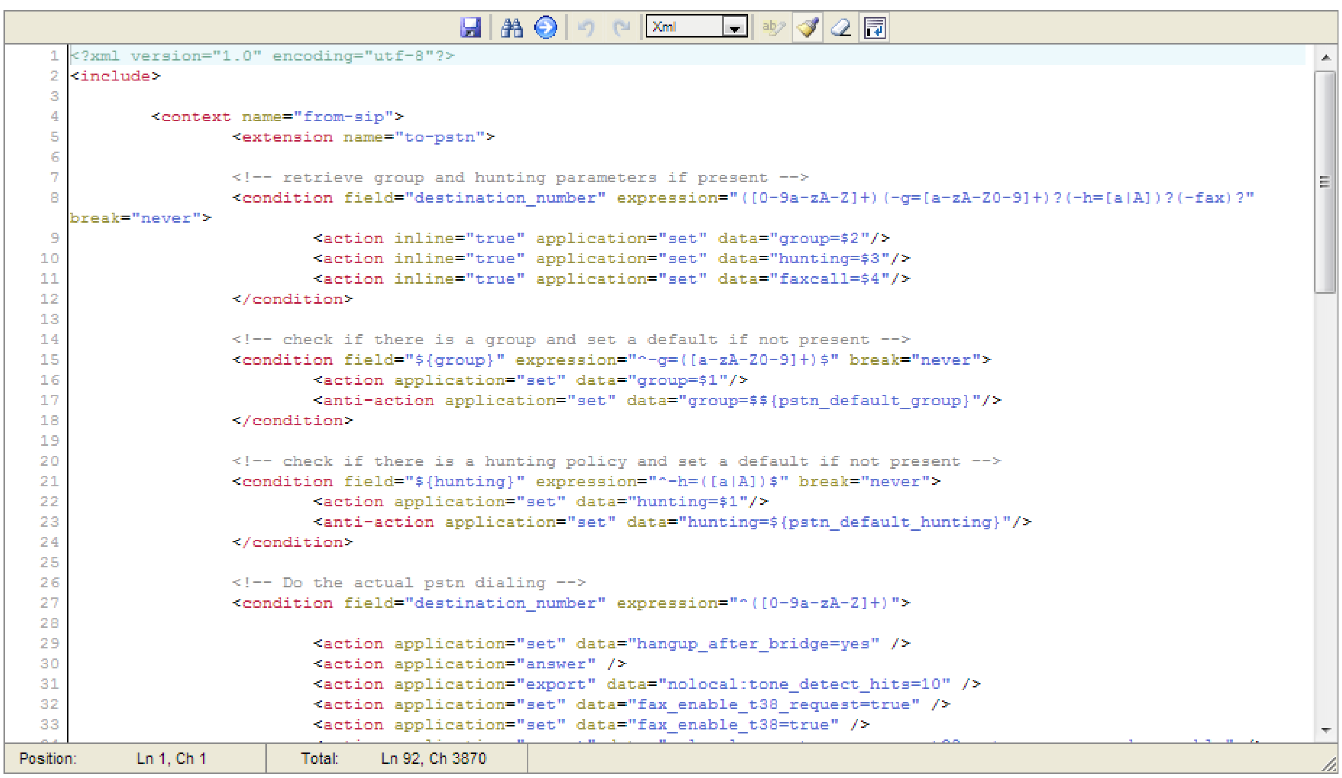
12. Dialplan

When a call is received in the NetBorder SS7 Gateway, the dialplan is fetched to retrieve the route information to find the outgoing call location.

- PSTN to SIP Dialplan
- SIP to PSTN Dialplan
- References

To access Dialplan configuration section

- Select **Dialplan** from side/top **Configuration** Menu
- Change a variable and Click on Save (Disk Icon)
- Proceed to Control Panel and Restart the VoIP Gateway.



```xml
<?xml version="1.0" encoding="utf-8"?>
<include>

        <context name="from-sip">
                <extension name="to-pstn">

                <!-- retrieve group and hunting parameters if present -->
                <condition field="destination_number" expression="([0-9a-zA-Z]+)(-g=[a-zA-Z0-9]+)?(-h=[a|A])?(-fax)?"
break="never">
                        <action inline="true" application="set" data="group=$2"/>
                        <action inline="true" application="set" data="hunting=$3"/>
                        <action inline="true" application="set" data="faxcall=$4"/>
                </condition>

                <!-- check if there is a group and set a default if not present -->
                <condition field="${group}" expression="^-g=([a-zA-Z0-9]+)$" break="never">
                        <action application="set" data="group=$1"/>
                        <anti-action application="set" data="group=$${pstn_default_group}"/>
                </condition>

                <!-- check if there is a hunting policy and set a default if not present -->
                <condition field="${hunting}" expression="^-h=([a|A])$" break="never">
                        <action application="set" data="hunting=$1"/>
                        <anti-action application="set" data="hunting=${pstn_default_hunting}"/>
                </condition>

                <!-- Do the actual pstn dialing -->
                <condition field="destination_number" expression="^([0-9a-zA-Z]+)">

                        <action application="set" data="hangup_after_bridge=yes" />
                        <action application="answer" />
                        <action application="export" data="nolocal:tone_detect_hits=10" />
                        <action application="set" data="fax_enable_t38_request=true" />
                        <action application="set" data="fax_enable_t38=true" />
```

Position: Ln 1, Ch 1 Total: Ln 92, Ch 3870

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

Dialplan is pre-configured for

- SIP to TDM and TDM to SIP Bridging.
  Section "from-sip" routes calls from SIP to PSTN/SS7
  Section "from-pstn" routes calls from PSTN/SS7 to SIP.

- H323 to TDM and TDM to H323 Bridging
  Section “from-h323” routes calls from H323 to PSTN

Dialplan Reload/Apply

Note that Dialplan can be modified in real time without the need to restart the gateway.
Once you Save the Dialplan, you will be prompted to Reload the gateway which will apply the changes without any service interrupt. All the currently established calls will not be affected. Only the newly established calls will start using the new dialplan rules.

PSTN to SIP Dialplan

```
<context name="from-pstn">
	<extension name="to-h323">
		<!-- handle the case where there might not be destination number at all -->
		<condition field="destination_number" expression="^(.{1,})$" break="never">
			<action application="set" data="destnumber=$1"/>
			<anti-action application="set" data="destnumber=unknown"/>
		</condition>

		<!-- Dial to the gateway user (it may ring multiple registrations, first answer wins) -->
		<condition field="destination_number" expression="^(.*)$">

			<action application="set" data="hangup_after_bridge=yes" />
			<action application="bridge" data="opal/h323:${destination_number}@${h323_remote_ip}"/>

			<!-- uncomment this if you want to dial to a fixed IP addr -->
			<!-- <action application="bridge" data="sofia/internal/${destnumber}@192.168.1.1"/> -->
		</condition>
	</extension>

	<extension name="to-sip">
		<!-- handle the case where there might not be destination number at all -->
		<condition field="destination_number" expression="^(.{1,})$" break="never">
			<action application="set" data="destnumber=$1"/>
			<anti-action application="set" data="destnumber=unknown"/>
		</condition>

		<!-- Dial to the gateway user (it may ring multiple registrations, first answer wins) -->
		<condition field="destination_number" expression="^(.*)$">

			<action application="set" data="hangup_after_bridge=yes" />
			<action application="set" data="tone_detect_hits=1" />
			<action application="set" data="fax_enable_t38_request=true" />
			<action application="set" data="fax_enable_t38=true" />
			<action application="set" data="execute_on_answer=t38_gateway peer cng" />

			<action application="set" data="sip_contact_user_replacement=${destnumber}"/>

			<action application="set" data="hangup_after_bridge=yes"/>

			<!—Bridge call to a registered SIP UA.  Comment if you want to bridge to IP -->
			<action application="bridge" data="${sofia_contact($${gwuser}@$${domain})}"/>
			<!-- Uncomment this if you want to dial to a fixed IP addr -->
			<!-- <action application="bridge" data="sofia/internal/${destnumber}@192.168.1.1"/> -->

			<action application="hangup" data="${originate_disposition}"/>

		</condition>
	</extension>
</context>
```

By default NSG is setup to accept SIP registrations. All calls coming from PSTN would be passed to a registered SIP User Agent.

If you would like NSG to send SIP requests to a specific address, you must uncomment the above line with blue arrow. And comment the line above it.
Then insert your own IP address instead of the 192.168.1.1 which is there as a example.

SIP to PSTN Dialplan

```
<context name="from-sip">
        <extension name="to-pstn">
                <!-- retrieve group and hunting parameters if present -->
                <condition field="destination_number" expression="([0-9a-zA-Z]+)(-g=[a-zA-Z0-9]+)?(-h=[a|A])?(-fax)?"
break="never">
                        <action inline="true" application="set" data="group=$2"/>
                        <action inline="true" application="set" data="hunting=$3"/>
                        <action inline="true" application="set" data="faxcall=$4"/>
                </condition>
                <!-- check if there is a group and set a default if not present -->
                <condition field="${group}" expression="^-g=([a-zA-Z0-9]+)$" break="never">
                        <action application="set" data="group=$1"/>
                        <anti-action application="set" data="group=$${pstn_default_group}"/>
                </condition>
                <!-- check if there is a hunting policy and set a default if not present -->
                <condition field="${hunting}" expression="^-h=([a|A])$" break="never">
                        <action application="set" data="hunting=$1"/>
                        <anti-action application="set" data="hunting=${pstn_default_hunting}"/>
                </condition>
                <!-- Do the actual pstn dialing -->
                <condition field="destination_number" expression="^([0-9a-zA-Z]+)">

                        <action application="set" data="hangup_after_bridge=yes" />
                        <action application="answer" />
                        <action application="export" data="nolocal:tone_detect_hits=10" />
                        <action application="set" data="fax_enable_t38_request=true" />
                        <action application="set" data="fax_enable_t38=true" />
                        <action application="export" data="nolocal:execute_on_answer=t38_gateway peer
ced_preamble" />

                        <action application="set" data="hangup_after_bridge=yes"/>
                        <action application="bridge" data="freetdm/${group}/${hunting}/$1"/>
                        <action application="hangup" data="${originate_disposition}"/>
                </condition>
        </extension>
</context>
```

Dialplan Syntax

There are several elements used to build an XML dialplan. In general, the dialplan groups logically similar functions and calling activities into a 'context'. Within a context are extensions, each with 'condition' rules and associated 'actions' to perform when the condition rules match.

The following is a sample dialplan to illustrate these concepts. We have left out the XML "wrapper" to help make the basic concepts more clear:

```
<context name="example">
  <extension name="500">
    <condition field="destination_number" expression="^500$">
      <action application="bridge" data="user/500"/>
    </condition>
  </extension>

  <extension name="501">
    <condition field="destination_number" expression="^501$">
      <action application="bridge" data="user/501"/>
      <action application="answer"/>
      <action application="sleep" data="1000"/>
      <action application="bridge" data="loopback/app=voicemail:default ${domain_name} ${dialed_extension}"/>
    </condition>
  </extension>
</context>
```

Each rule is processed in order until you reach the action tag which tells NSG what action to perform. You are not limited to only one condition or action tag for a given extension.

In our above example, a call to extension 501 rings the extensions. If the user does not answer, the second action answers the call, and following actions delay for 1000 milliseconds (which is 1 second) and connect the call to the voicemail system.

Context

Contexts are a logical grouping of extensions. You may have multiple extensions contained within a single context.

The context tag has a required parameter of 'name'. There is one reserved name, any, which matches any context. The name is used by incoming call handlers (like the [Sofia] SIP driver) to select the dialplan that runs when it needs to route a call. There is often more than one context in a dialplan.

A fully qualified context definition is shown below. Typically you'll not need all the trimmings, but they are shown here for completeness.

```
<?xml version="1.0"?>
  <document type="freeswitch/xml">
    <section name="dialplan" description="Regex/XML Dialplan">
    <!-- the default context is a safe start -->
      <context name="default">
        <!-- one or more extension tags -->
      </context>
      <!-- more optional contexts -->
    </section>
  </document>
```

Extensions

Extensions are destinations for a call. This is the meat of NSG routing dialed numbers. They are given a name and contain a group of conditions, that if met, will execute certain actions.

A 'name' parameter is required: It must be a unique name assigned to an extension for identification and later use.

For example:

```
<extension name="Your extension name here">
    <condition(s)...
        <action(s) .../>
    </condition>
</extension>
```

NOTE: Typically when an extension is matched in your dialplan, the corresponding actions are performed and dialplan processing stops. An optional `continue` parameter allows your dialplan to continue running.

```
<extension name="500" continue="true">
```

Conditions

Dialplan conditions are typically used to match a destination number to an extension. They have, however, much more power than may appear on the surface.
NSG has a set of built-in variables used for testing. In this example, the built-in variable destination_number is compared against the regular expression ^500$.
This comparison is 'true' if <destination_number> is set to 500.

```
<extension name="500">
   <condition field="destination_number" expression="^500$">
      <action application="bridge" data="user/500"/>
   </condition>
</extension>
```

Each condition is parsed with the Perl Compatible Regular Expression library. (go here for PCRE syntax information).

If a regular expression contains any terms wrapped in parentheses, and the expression matches, the variables $1,$2..$N will be set to the matching contents within the parenthesis, and may be used in subsequent action tags within this extension's block.

For example, this simple expression matches a four digit extension number, and captures the last two digits into $1.

```
<condition field="destination_number" expression="^\d\d(\d\d)$">
   <action application="bridge" data="sofia/internal/$1@example.com"/>
</condition>
```

A destination number of 3425 would set $1 to 25 and then bridge the call to the phone at 25@example.com

Multiple Conditions (Logical AND)

You can emulate the logical AND operation available in many programming languages using multiple conditions. When you place more than one condition in an extension, *all* conditions must match before the actions will be executed. For example, this block will only execute the actions if the destination number is 500 *AND* it is Sunday.

```
<condition field="destination_number" expression="^500$"/>
<condition wday="1">
    action(s)...
</condition>
```

Keep in mind that you must observe correct XML syntax when using this structure. Be sure to close all conditions *except the last one* with `/>`. The last condition contains the final actions to be run, and is closed on the line after the last action.
By default, if any condition is false, NSG will move on to the anti-actions or the next extension without even evaluating any more conditions.

Multiple Conditions (Logical OR, XOR)

It is possible to emulate the logical OR operation available in many programming languages, using multiple conditions. In this situation, if one of the conditions matches, the actions are executed. For example, this block executes its actions if the destination number is 501 *OR* the destination number is 502.

```
<condition field="destination_number" expression="^501|502$">
  action(s)...
</condition>
```

This method works well if your OR condition is for the same field. However, if you need to use two or more different fields then use the new **regex** syntax

```
<extension name="Regex OR example 1" continue="true">
  <condition regex="any">
    <!-- If either of these is true then the subsequent actions are added to execute list -->
    <regex field="caller_id_name" expression="Some User"/>
    <regex field="caller_id_number" expression="^1001$"/>
    <action application="log" data="INFO At least one of the conditions matched!"/>
    <!-- If *none* of the regexes is true then the anti-actions are added to the execute list -->
    <anti-action application="log" data="WARNING None of the conditions matched!"/>
  </condition>
</extension>
```

Using this method it becomes easier to match the caller's name OR caller ID number and execute actions whether either is true.
A slightly more advanced use of this method is demonstrated here:

```
<extension name="Regex OR example 2" continue="true">
  <condition regex="any" break="never">
    <regex field="caller_id_name" expression="^Michael\s*S?\s*Collins"/>
    <regex field="caller_id_number" expression="^1001|3757|2816$"/>
    <action application="set" data="calling_user=mercutioviz" inline="true"/>
    <anti-action application="set" data="calling_user=loser" inline="true"/>
  </condition>

  <condition>
    <action application="answer"/>
```

```
      <action application="sleep" data="500"/>
      <action application="playback" data="ivr/ivr-welcome_to_freeswitch.wav"/>
      <action application="sleep" data="500"/>
    </condition>

    <condition field="${calling_user}" expression="^loser$">
      <action application="playback" data="ivr/ivr-dude_you_suck.wav"/>
      <anti-action application="playback" data="ivr/ivr-dude_you_rock.wav"/>
    </condition>
  </extension>
  <extension name="Regex XOR example 3" continue="true">
    <condition regex="xor">
      <!-- If only one of these is true then the subsequent actions are added to execute list -->
      <regex field="caller_id_name" expression="Some User"/>
      <regex field="caller_id_number" expression="^1001$"/>
      <action application="log" data="INFO Only one of the conditions matched!"/>
      <!-- If *none* of the regexes is true then the anti-actions are added to the execute list -->
      <anti-action application="log" data="WARNING None of the conditions matched!"/>
    </condition>
  </extension>
```

Basically, for this new syntax you can have a condition to have a "regex" attr instead of "field" and "expression" etc. When there is a "regex" attr, that means you plan to have one or more <regex> tags that are similar to the condition tag itself that it has field and expression in it.

The value of the "regex" attr is either "all" or "any" or "xor indicating if all expressions must match or just any expression or only one must match(xor) . If it's set to "any" it will stop testing the regex tags as soon as it finds one match, if it is set to "all", it will stop as soon as it finds one failure.

From there it will behave like a normal condition tag either executing the actions or anti-actions and breaking based on the "break" attr.

The basic difference here is once there is a "regex" attr, the <regex> tags parsed for "all" or "any" take the place of the single "field" and "condition"

NOTE: Also, if any captures are done in the "expression" attrs of a <regex> tag, only the data from the newest capture encountered will be considered in the $n expansion or FIELD_DATA creation. In addition, you can set DP_REGEX_MATCH_1 .. DP_REGEX_MATCH_N to preserve captures into arrays.

```
  <extension name="Inbound_external">
```

```
  <condition regex="any">
    <regex field="${sip_from_host}" expression="domainA"/>
    <regex field="${sip_from_uri}" expression="1234567890@domainB"/>
    <regex field="${sip_from_uri}" expression="user@domainC"/>
    <regex field="caller_id_name" expression="^(John Smith)$"/>
    <regex field="caller_id_number" expression="^(55512341)|(55512342)|(55512343)$"/>

    <action application="set" data="domain_name=domainZ"/>
    <action application="transfer" data="${destination_number} XML domainZ"/>
  </condition>
</extension>
```

This is another example to show that all regex conditions must be true, then the action will get executed; otherwise, the anti-action will. This is the same logic as follows:

```
IF (cond1 AND cond2 AND cond3) THEN
 do actions
ELSE
 do other actions
ENDIF
```

Basically, the <condition regex="all"> tells the parser, "Hey, execute the <action>'s only if all regexes PASS, otherwise execute any <anti-action>'s".

```
 <condition regex="all">
<regex field="${sip_gateway}" expression="^${default_provider}$"/>
<regex field="${emergency_call}" expression="^true$"/>
<regex field="${db(select/emergency/autoanswer)}" expression="^1$"/>

<!-- the following actions get executed if all regexes PASS -->
 <action application="set" data="call_timeout=60"/>
 <action application="set" data="effective_caller_id_name=${regex(${caller_id_name}|^Emerg(_.*)$|Auto%1)}"/>
 <action application="set" data="autoanswered=true"/>
 <action application="bridge" data="user/1000@${domain_name},sofia/gateway/1006_7217/${mobile_number}"/>

<!-- the following anti-actions are executed if any of the regexes FAIL -->
 <anti-action application="set" data="effective_caller_id_name=${regex(${caller_id_name}|^Emerg(_.*)$|NotAuto%1)}"/>
 <anti-action application="set" data="call_timeout=30"/>
 <anti-action application="set" data="autoanswered=false"/>
```

```
  <anti-action application="bridge" data="user/1000@${domain_name},sofia/gateway/1006_7217/${mobile_number}"/>
</condition>
```

Complex Condition/Action Rules

Here is a more complex example, performing time-based routing for a support organization. The user dials extension 1100. The actual support extension is 1105 and is staffed every day from 8am to 10pm, except Friday, when it is staffed between 8am and 1pm. At all other times, calls to 1100 are sent to the support after-hours mailbox.

```
<extension name="Time-of-day-tod">
   <!--if this is false, FreeSWITCH skips to the next *extension*.-->
   <condition field="destination_number" expression="^1100$" break="on-false"/>

   <!--Don't bother evaluating the next condition set if this is true.-->
   <condition wday="6" hour="8-12" break="on-true">    <!--Fri, 8am-12:59pm-->
      <action application="transfer" data="1105 XML default"/>
   </condition>

   <condition wday="1-5" hour="8-21" break="on-true">   <!--Sunday-Thursday, 8am-9:59pm-->
      <action application="transfer" data="1105 XML default"/>
   </condition>

   <condition> <!--this is a catch all, sending the call to voicemail at all other times. -->
      <action application="voicemail" data="default ${domain} 1105"/>
   </condition>
</extension>
```

In this example, we use the `break=never` parameter to cause the first condition to 'fall-through' to the next condition no matter if the first condition is true or false. This is useful to set certain flags as part of extension processing. This example sets the variable `begins_with_one` if the destination number begins with 1.

```
<extension name="break-demo">
 <!-- because break=never is set, even when the destination does not begin
      with 1, we skip the action and keep going -->
 <condition field="destination_number" expression="^1(\d+)$" break="never">
   <action application="set" data="begins_with_one=true"/>
 </condition>

 <condition field="destination_number" expression="^(\d+)$">
    ...other actions that may query begins_with_one...
 </condition>
</extension>
```

Variables

Condition statements can match against channel variables, or against an array of built in variables.

Built-In Variables

The following variables, called 'caller profile fields', can be accessed from condition statements directly:

- **context** Why can we use the context as a field? Give us examples of usages please.
- **rdnis** Redirected Number, the directory number to which the call was last presented.
- **destination_number** Called Number, the number this call is trying to reach (within a given context)
- **dialplan** Name of the dialplan module that are used, the name is provided by each dialplan module. Example: XML
- **caller_id_name** Name of the caller (provided by the User Agent that has called us).
- **caller_id_number** Directory Number of the party who called (caller) -- can be masked (hidden)
- **ani** Automatic Number Identification, the number of the calling party (caller) -- cannot be masked
- **aniii** The type of device placing the call ANI2
- **uuid** Unique identifier of the current call? (looks like a GUID)
- **source** Name of the FreeSWITCH module that received the call (e.g. PortAudio)
- **chan_name** Name of the current channel (Example: PortAudio/1234). Give us examples when this one can be used.
- **network_addr** IP address of the signaling source for a VoIP call.
- **year** Calendar year, 0-9999
- **yday** Day of year, 1-366
- **mon** Month, 1-12 (Jan = 1, etc.)
- **mday** Day of month, 1-31
- **week** Week of year, 1-53
- **mweek** Week of month, 1-6
- **wday** Day of week, 1-7 (Sun = 1, Mon = 2, etc.) or "sun", "mon", "tue", etc.
- **hour** Hour, 0-23
- **minute** Minute (of the hour), 0-59
- **minute-of-day** Minute of the day, (1-1440) (midnight = 1, 1am = 60, noon = 720, etc.)
- **time-of-day** Time range formatted: hh:mm[:ss]-hh:mm[:ss] (seconds optional) Example: "08:00-17:00"
- **date-time** Date/time range formatted: YYYY-MM-DD hh:mm[:ss]~YYYY-MM-DD hh:mm[:ss] (seconds optional, note tilde between dates) Example: 2010-10-01 00:00:01~2010-10-15 23:59:59

For example:

```
<condition field="network_addr" expression="^192\.168\.1\.1$"/>  <!-- network address=192.168.1.1 >
<condition mon="2">   <!-- month=February -->
```

Caller Profile Fields vs. Channel Variables

One thing that may seem confusing is the distinction between a caller profile field (the built-in variables) and a channel variable.
Caller profile fields are accessed like this:

```
<condition field="destination_number" attributes...>
```

While channel variables are accessed like this:

```
<condition field="${sip_has_crypto}" attributes...>
```

Please take note of the **${variable_name}** syntax. Channel variables may also be used in action statements. In addition, API functions can be called from inside a condition statement to provide dynamic data.

For example, you can use the **cond** API:

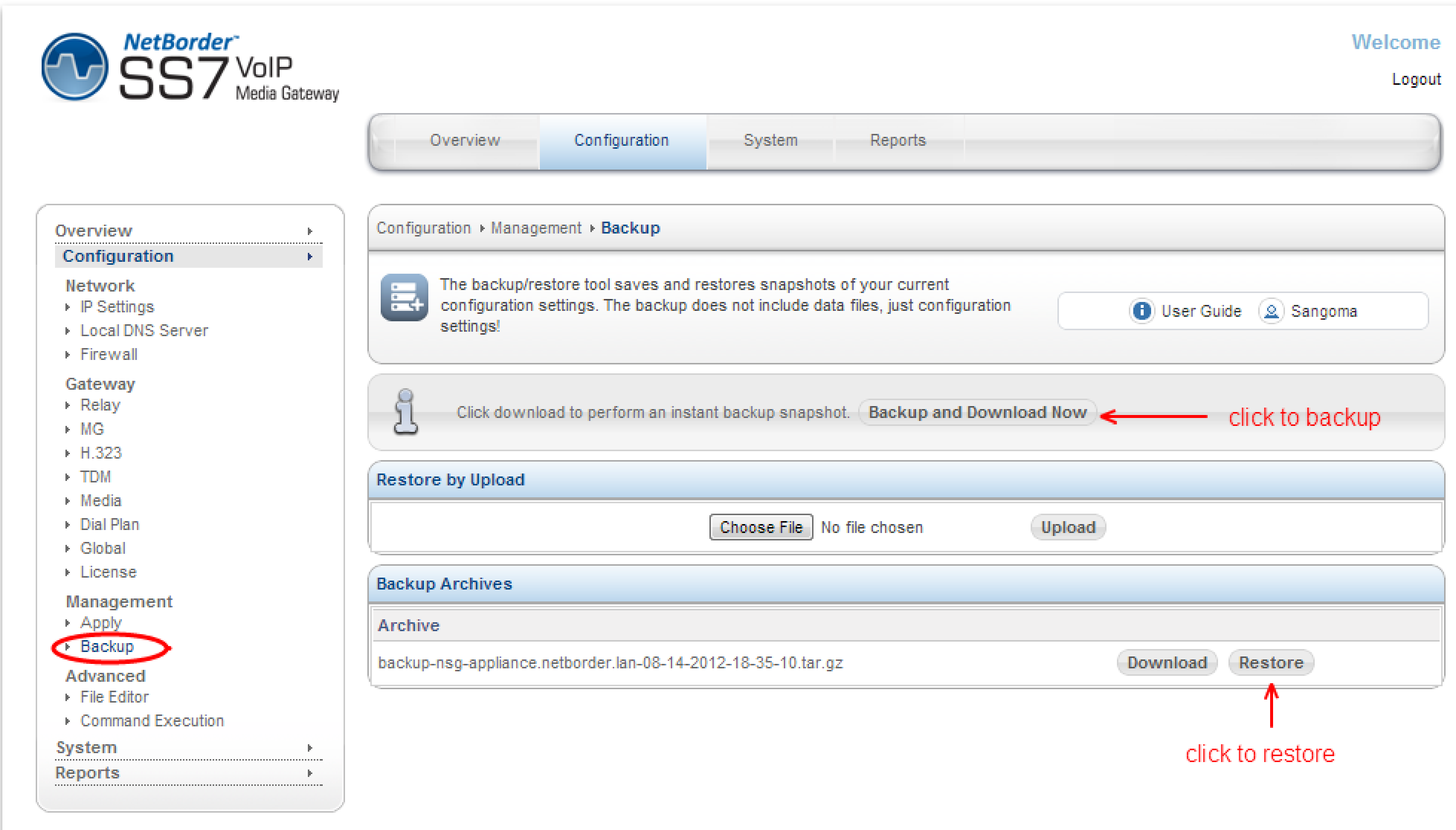
```
<condition field="${cond(${my_var} > 12 ? YES : NO)}" expression="^YES$">
   <action application="log" data="INFO ${my_var} is indeed greater than 12"/>
</condition>
```

This example tests ${my_var}. If it is more than 12, "YES" is returned. Otherwise "NO" is returned. The condition tests the results for "YES" and logs the resulting message to the NSG log.

13. Backup Restore System

Appliance configuration can be backed up to a zipped file.
Appliance can be restored from a same file.

- Select **Backup** from side/top **Configuration** Menu



14. Factory Reset & Reboot

Factory Reset

- Find a power button in front of the NSG Server
- Press the power button repeatedly every 1 sec for 5 sec.
- After 5 power button presses within 5sec the NSG System will revert to factory defaults.
- The TDM at the back of the server rear panel will flash three times to indicate that the factory reset has been performend

Server Reboot

- Find a power button in front of the NSG Server
- Press the power button twice with more than 2sec delay in between..
- When there were 5 power button presses within 5sec the NSG System will revert to factory defaults.

Server Shutdown

- Find a power button in front of the NSG Server
- Press the power button and hold it until machine shutsdown.

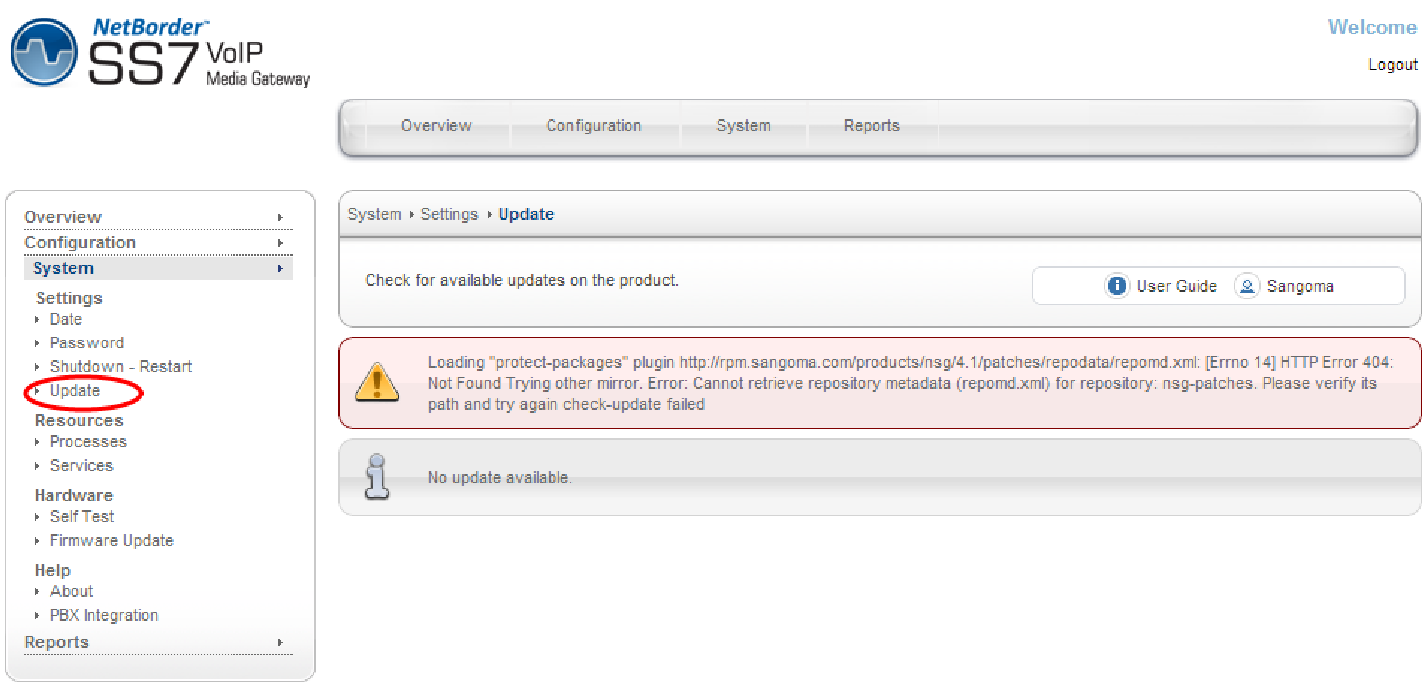
16. Upgrade

User has two choices when upgrading NSG system.

- WebUI Update Page
- Manuall Console Update via SSH

WebUI System Update

- Select **Update** from side/top **System** Menu
- Review available packages for upgrade.
- Proceed with the upgrade process



Console SSH Update

NSG product uses Linux RPM as part of its package management system.

- Download new NSG RPM version

- Stop NSG services
  - User the GUI Control Panel
  - Alternatively run:
    - services nsg stop
    - services nsg-webui stop

- Install new package
  - rpm –Uvh nsg-4.3.1.rpm

- Restart NSG services
  - Use the GUI Control Panel
  - Alternatively run:
    - services nsg-webui start
    - services nsg start

NOTE
Using NSG console to upgrade the system is very powerful, as the process can be scripted and centralized. This way all NSG appliances in the files can be upgraded from a single upgrade machine in the NOC.

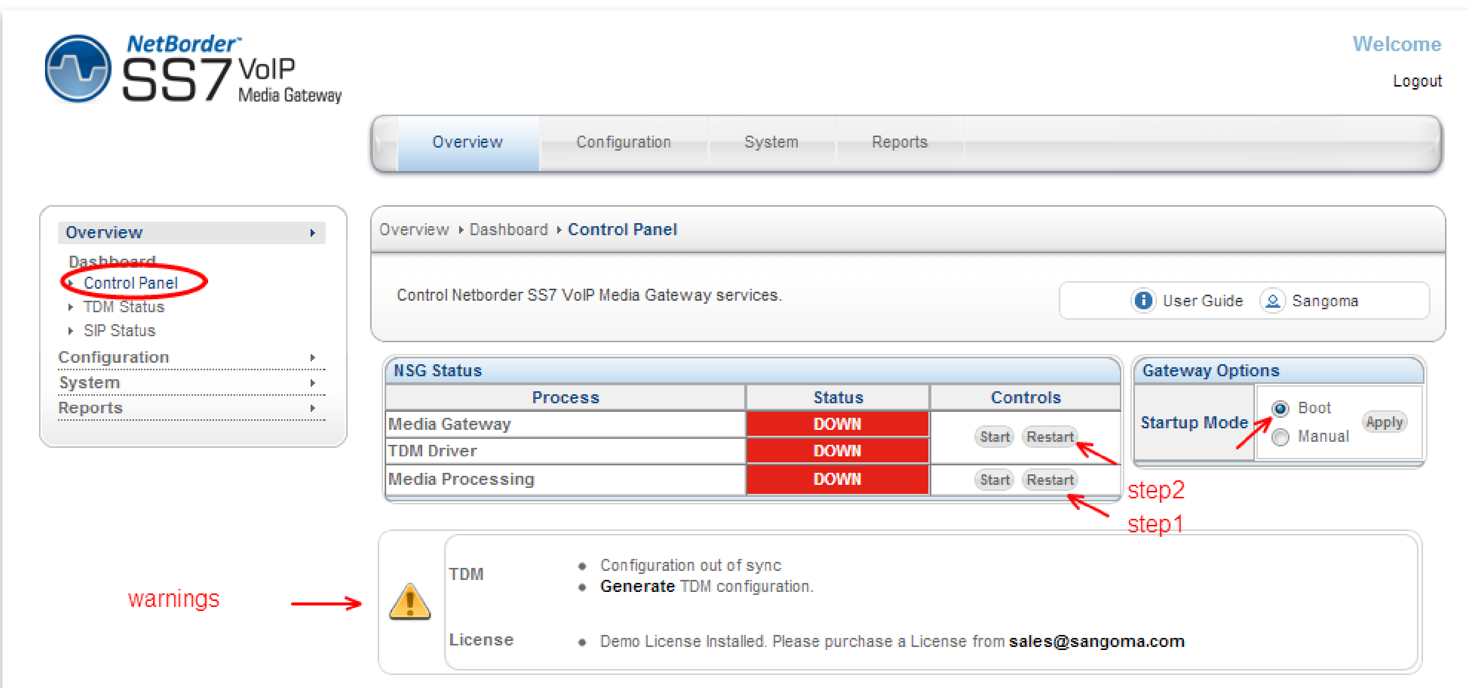
17. Operations

Starting the Gateway

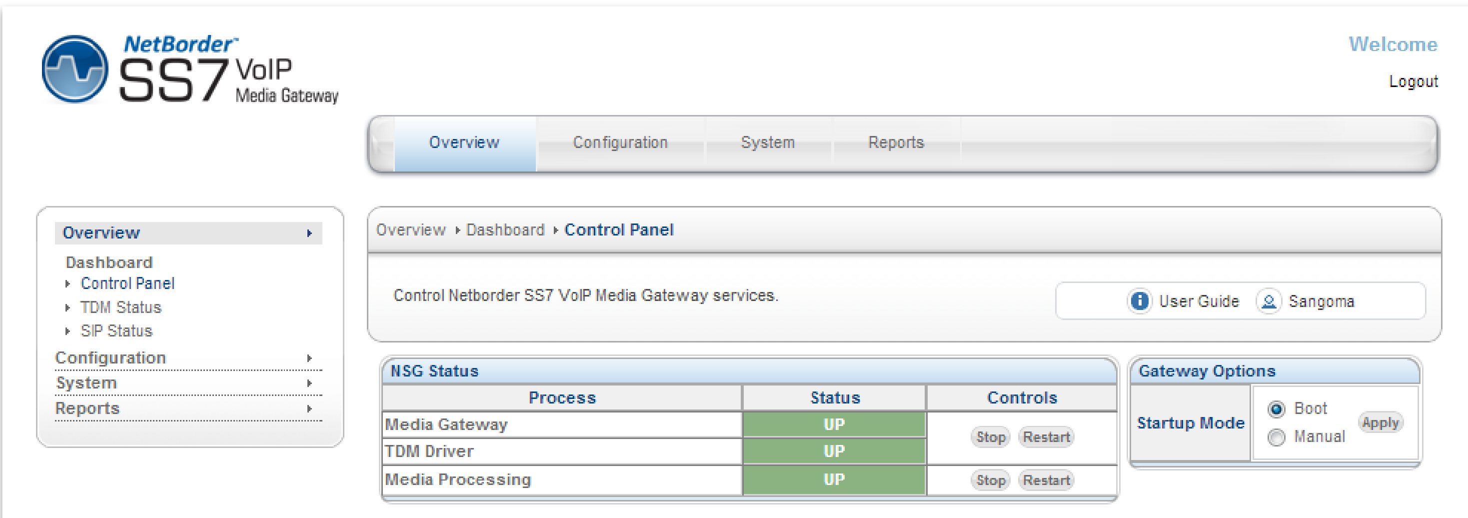
- Select **Control Panel** from side/top **Overview** Menu
- Confirm that warnings are clear
- Start the Media Processing First
  - Media Processing will start the Transcoding resources.
  - Note that Media Processing is optional

- Start the Media Gateway Second.
  - Media Gatway will start
  - TDM Hardware Spans (T1/E1 ports)
  - Netborder SS7 to VoIP Gateway Software

- Confirm that the **boot** button is selected.
  - This will confirm that gateway starts on reboot.



- When the Gateway starts successfully the green status bar will appear.
- System is now running.



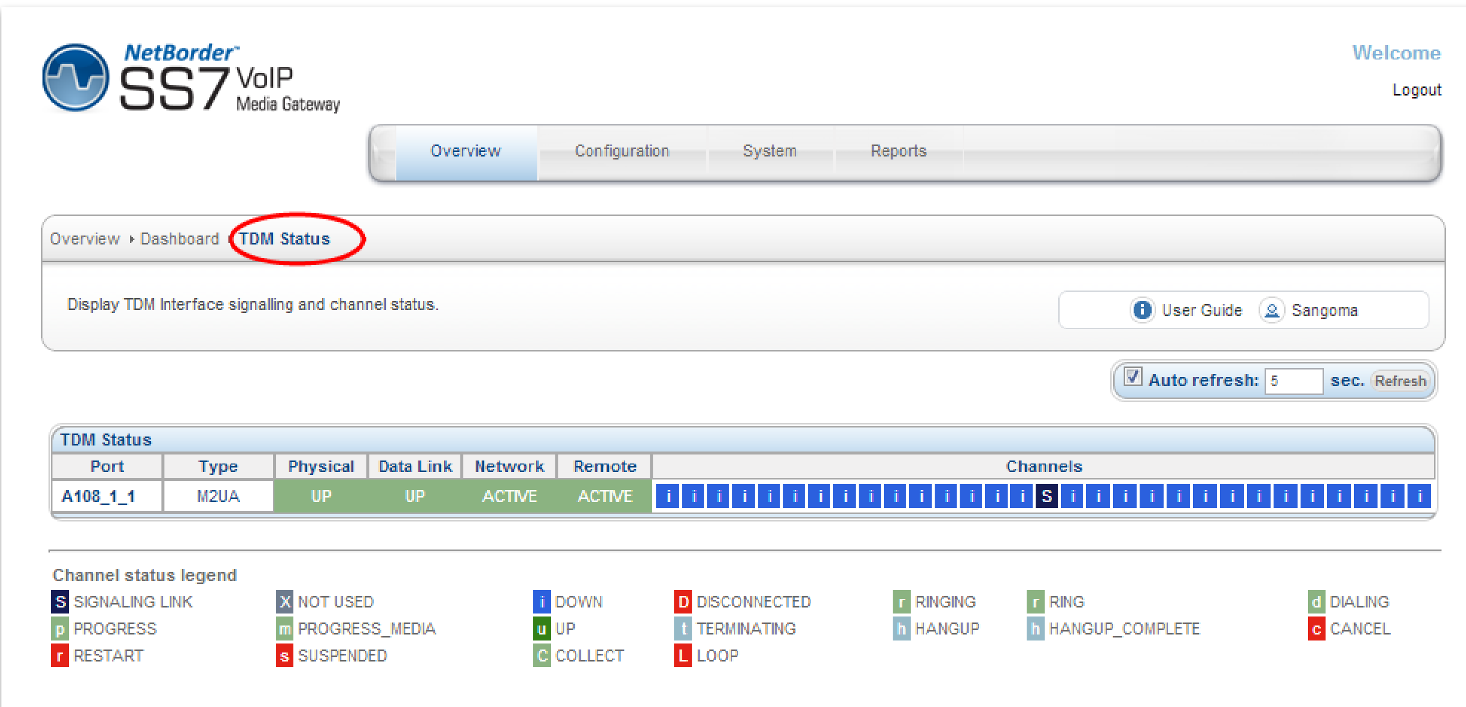
NOTE
Before attempting to pass traffic through the gateway, proceed to **TDM Status** to check the state of the NSG gateway.   There is no point of attempting calls while the status of the gateway protocol is down.

Gateway Status

Megaco/M2UA

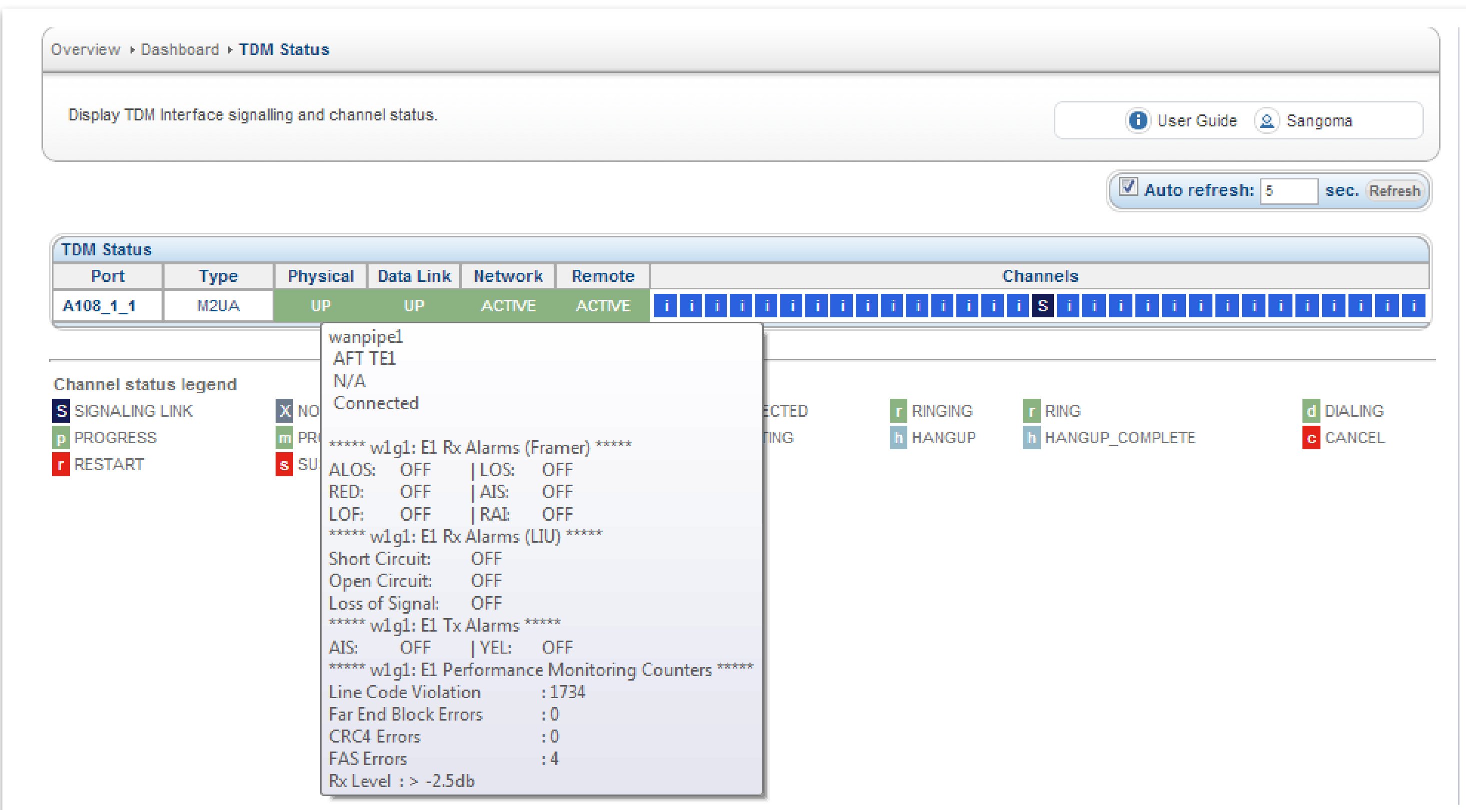
The TDM Stats page displays the unified status of all NSG components

- Select **TDM Stats** from side/top **Overview** Menu



Field Name	*Description*
Port	Physical Port number. Identifies the hardware resource and T1/E1 port number. The T1/E1 port number relates to the T1/E1 board.
Type	Signaling Type In this example we see: M2UA
Physical	Physical T1/E1 layer status. Hover the mouse over the Physical status section (green) to display detailed T1/E1 alarms and status.

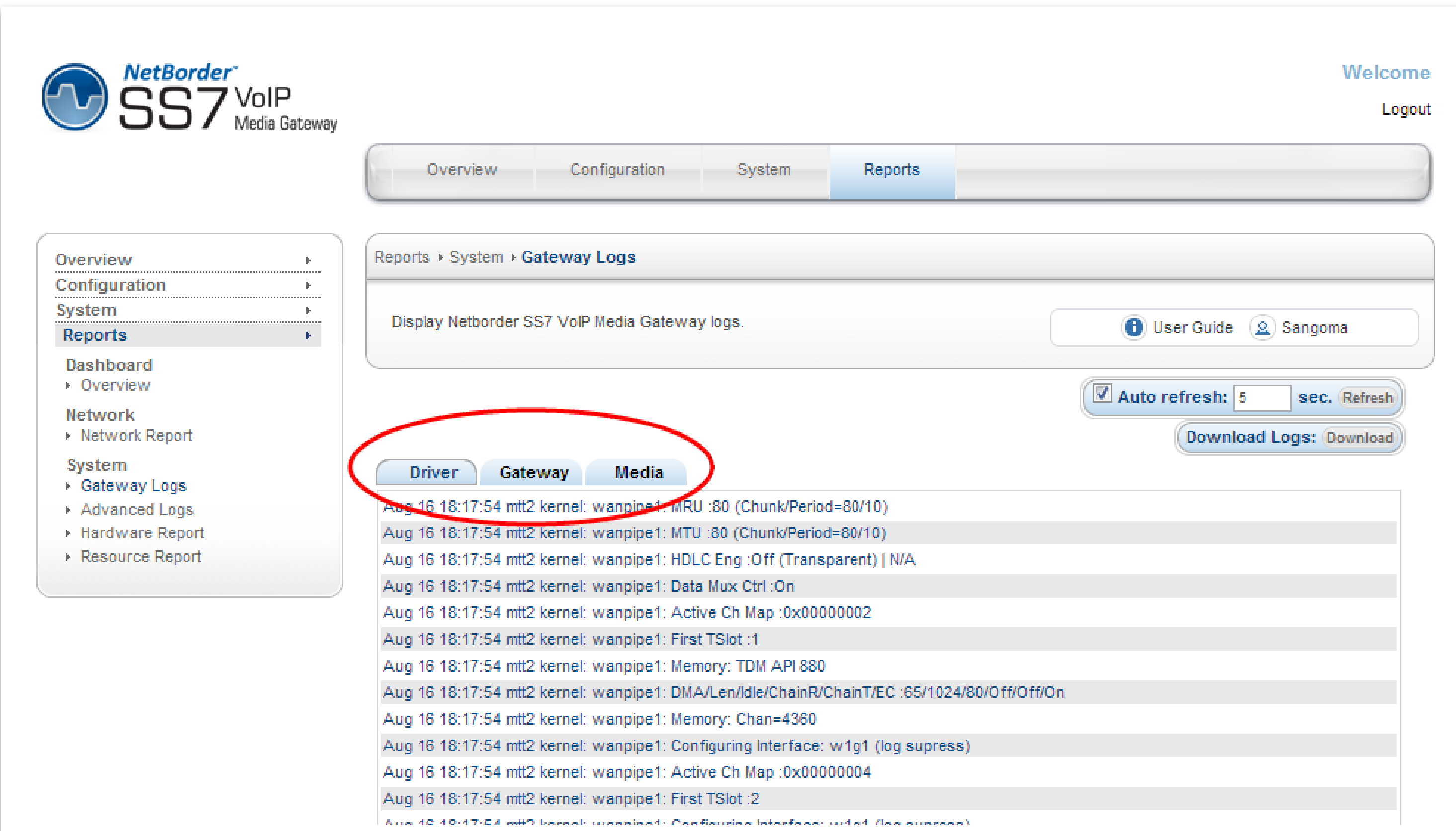
Data Link	MTP2 Link Layer status. Hover the mouse over the UP and a popup will display detailed MTP2 status
Network	M2UA Link Layer status Hover the mouse over the UP and a popup will display detailed M2UA status
Remote	Remote MGC Megaco Peer status. This indicates that MG is connected to the MGC Megaco profile. Hover the mouse over the UP and a popup will display detailed Megaco Peer status
Channels	If Megaco link state is IN-SERVICE Channel is blue - down If Megaco link state is OUT-OF-SERVICE Channel is red – down If channel is in use Channel is green – up Hover over each channel for more detailed data.



18. Reports

Gateway Logs

- Select **Gateway Logs** from side/top **Reports** Menu



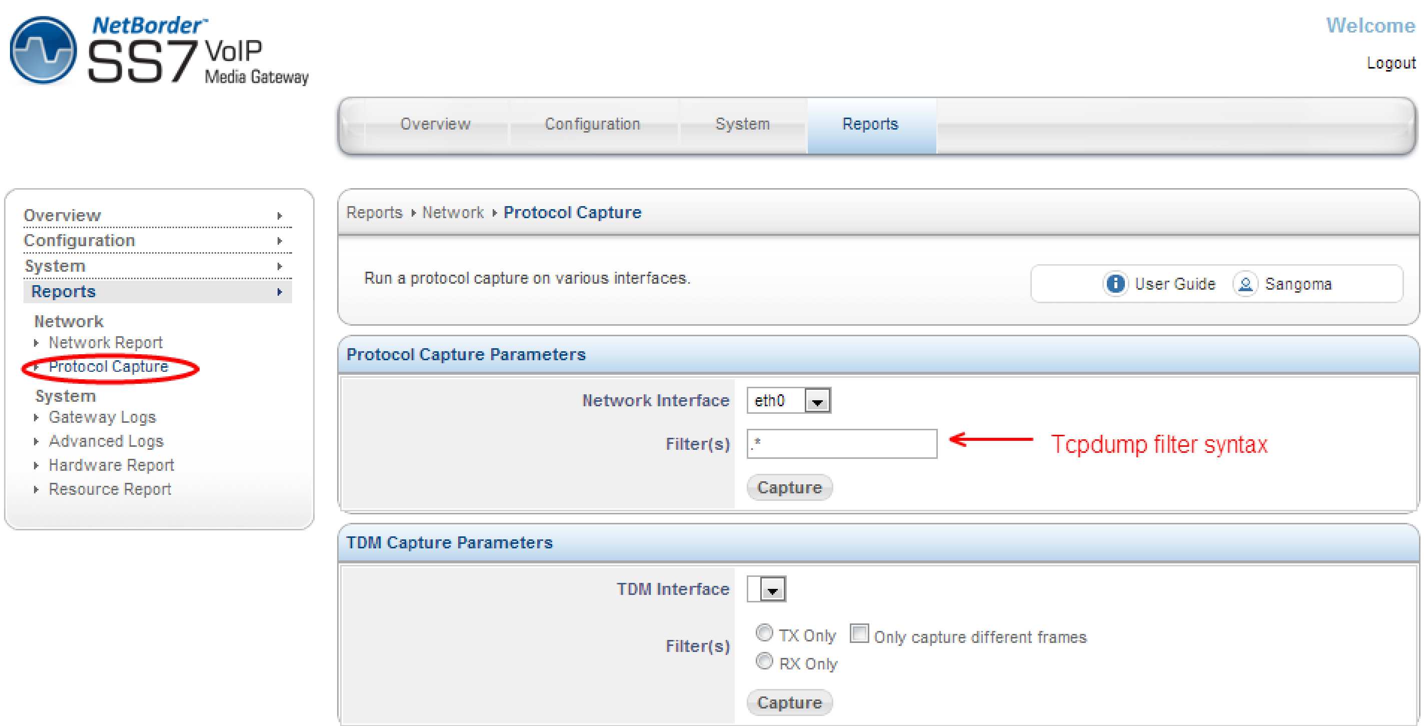
NOTE
All error events will be displayed in RED for easy identification.

Log	Description
Driver	TDM device driver log. All errors will be identified in RED This log will show • TDM Driver startup sequence • TDM T1/E1 connection/disconnection • TDM Driver general errors • System errors • OS Errors
Gateway	SS7 to VoIP Gateway log All errors will be identified in RED This log will show • Gateway startup sequence • Gateway startup errors • Gateway run time errors and warnings
Media	Media Transcoding log All errors will be identified in RED This log will show • Media Transcoding server startup sequence • Media startup errors • Media transcoding run time errors and warnings

Packet Capture

The packet capture page captures network traffic from Ethernet interface, TDM interface or both.

- Select **Packet Capture** from side/top **Reports** Menu
- Filter
  - Default filter will capture all packets on the Ethernet device
- Select Capture to start capturing
- Wait…
- Select Stop Capture when Capture done
- Download Link with capture pcap file is ready for download.



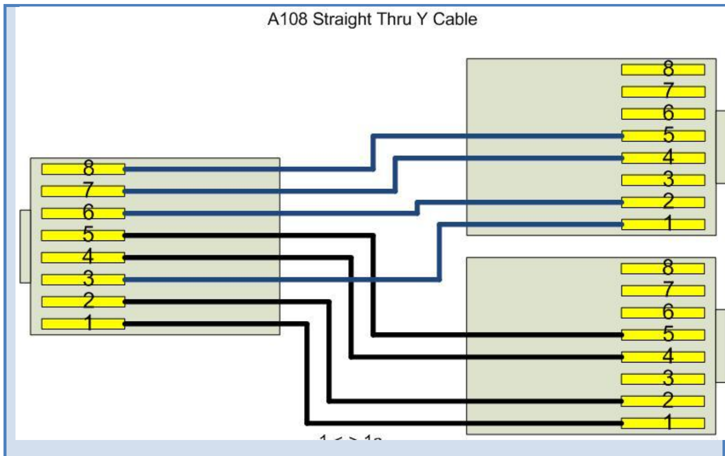
19. Cable Pinouts: T1/E1

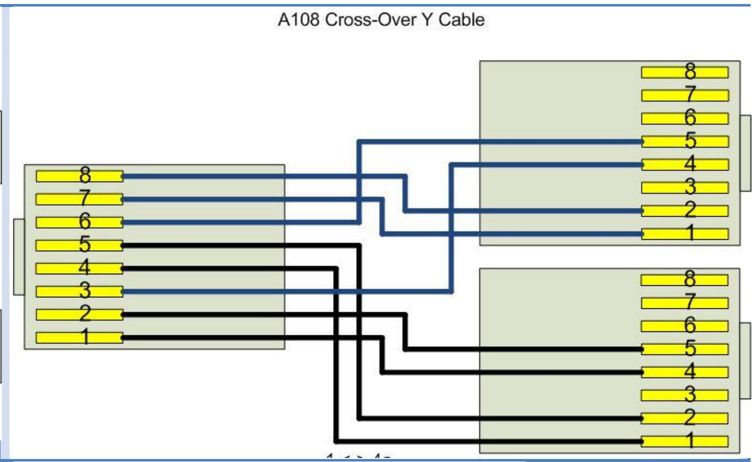
NSG Appliance utilizes Sangoma TDM T1/E1 digital board adapters.

- A101D / A101DE – 1-port E1/T1
- A102D / A102DE – 2-port E1/T1
- A104D / A104DE – 4-port E1/T1
- A108D / A108DE – 8-port E1/T1*

A108D Port Information	
The A108D card has dual purpose RJ45 connector, as it provides access to two T1/E1 ports from a single RJ45 Female connector. **NOTE** **There are two LED per RJ45 connector.**	

A108D Straight Cable	*A108D Cross Over – Back-to-Back Cable*
Y Cable for A108 to 2 separate T1/E1 (straight). **This is to connect the A108 against lines from the Telco.**	Y Cable for A108 to 2 separate T1/E1 (cross). This is to connect the A108 against another T1/E1 card.
A = port N; B = port N + 4 **1 <-> 1A [Rx ring]** **2 <-> 2A [Rx tip]** **3 <-> 1B** **4 <-> 4A [Tx ring]** **5 <-> 5A [Tx tip]** **6 <-> 2B** **7 <-> 4B** **8 <-> 5B**	A = port N; B = port N + 4 1 <-> 4A 2 <-> 5A 3 <-> 4B 4 <-> 1A 5 <-> 2A 6 <-> 5B 7 <-> 1B 8 <-> 2B





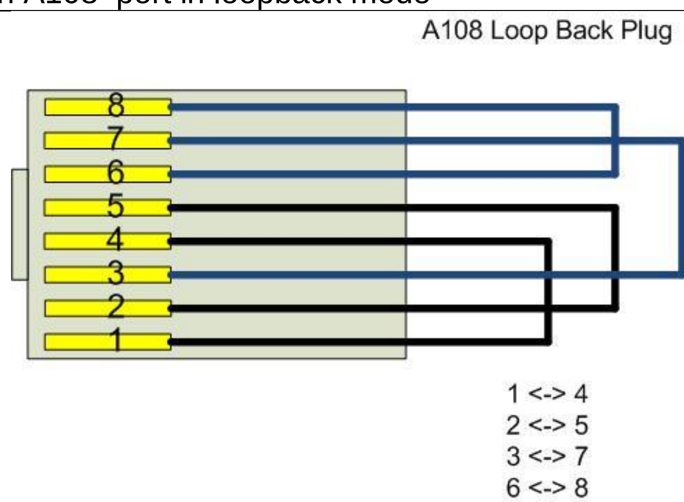
T1/E1 "Portsplitter" Cable
T1/E1 Split Cable for the A108
Standard | ROHS: Yes | Length: 6'

SKU: CABL-630

A108D Loop Back Cable

This is to connect an A108 port in loopback mode

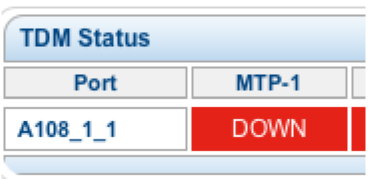
1 <-> 4
2 <-> 5
3 <-> 7
4 <-> 1
5 <-> 2
6 <-> 8
7 <-> 3
8 <-> 6



20. Troubleshooting

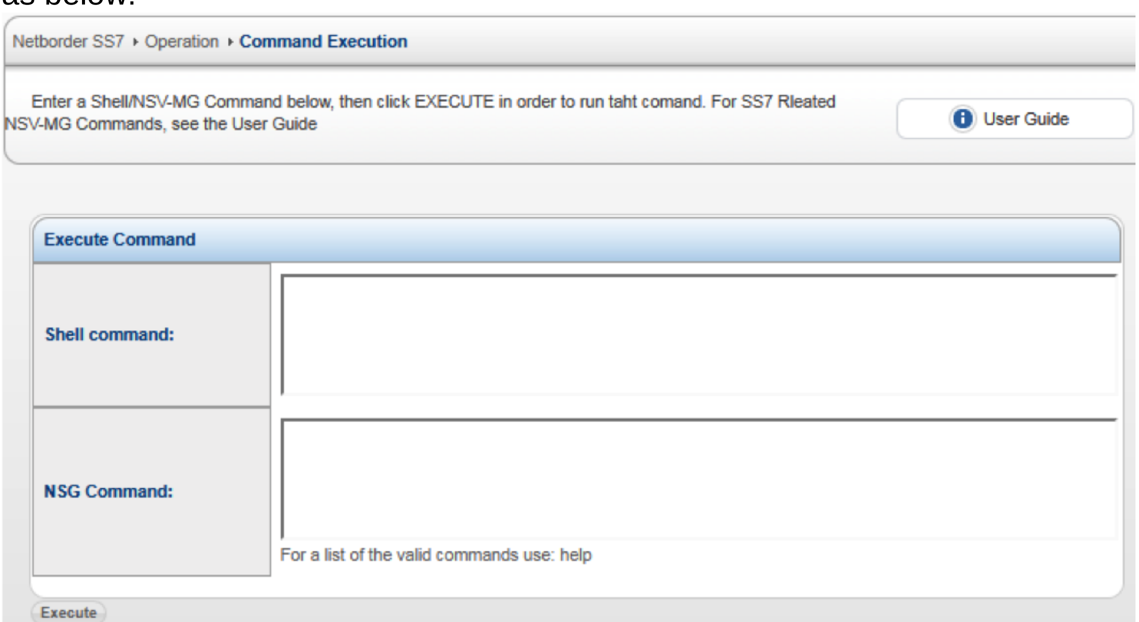
Physical Layer

The first step in troubleshooting any connectivity issue is troubleshooting the physical layer. Identifying whether a user has a physical layer issue is by using the TDM Status page and checking the MTP-1 column. If the column is listed as "DOWN" for that particular port, you must proceed with troubleshooting the physical layer.



When the physical layer is down, all layers above the physical layer will also be in a "DOWN" or "TRYING" state. In order to start troubleshooting, the user must proceed to the "Command Execution" page, which is located under the "Operation" column.

Once the user clicks on the Command Execution link, they will be presented with the following screen as below:



The best way to troubleshoot physical layer issues is through the shell command option. Below is a list of commands that can be run within the shell command section to help diagnose issues:

Linux Commands

- **ifconfig**
  Displays all network interfaces
  Sangoma interfaces start with "w" eg: "w1g1" for span1, "w2g1" for span2 ...
- **cat /proc/interrupts**
  Displays the interrupt status for all cpu and devices
- **hdparm -t /dev/sda**
  Check for disk access speed. If speeds are less than 10MBps it could indicated that motherboard/chipset is not supported by Linux kenrel.
- **vmstat**
  Outputs system load information
  in = interrupt per timeout
  cs = context switching
  us = user load
  sy = system load (kernel)
  id = idle

Sangoma TDM Driver related commands

- wanpipemon -i wXg1 -c Ta (where X is the span number in question. Can also be found using ifconfig)
  Output low level T1/E1 Alarms
- wanrouter status
  Output wanpipe physical status statistics

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

Wanpipe Port Status

The first step in debugging physical layer issues would be to check whether wanrouter status reports the line "Connected" or "Disconnected". To do this, within the "Shell Command" textbox, enter the command "wanrouter status". It will return a result like the one below:

-> **wanrouter status**

Shell Command

```
Devices currently active:
        wanpipe1


Wanpipe Config:

Device name | Protocol Map | Adapter  | IRQ | Slot/IO | If's | CLK | Baud rate |
wanpipe1    | N/A          | A101/1D/A102/2D/4/4D/8| 169 | 4        | 1    | N/A | 0          |

Wanrouter Status:

Device name | Protocol | Station | Status        |
wanpipe1    | AFT TE1  | N/A     | Disconnected  |
```

All the devices running on a NSG system will be listed as a "wanpipe" device. In this example, "wanpipe1" is being reported as "Disconnected", which tells us that the physical layer is in fact in a "DOWN" state.

Wanpipe Port T1/E1 Alarms

The next step would be to check where the issue lies.

To do this, the user would need to run the command

- wanpipemon -i wXg1 -c Ta
  (where X stands for the wanpipe number).

In this example, "wanpipe1" is in a disconnected state, therefore the interface name would be "w1g1".

The command returns an output similar to the one below:

-> wanpipemon -i w1g1 -c Ta

Shell Command

```
***** w1g1: E1 Rx Alarms (Framer) *****

ALOS:    OFF      | LOS:   ON
RED:     ON       | AIS:   OFF
LOF:     ON       | RAI:   OFF

***** w1g1: E1 Rx Alarms (LIU) *****

Short Circuit:  OFF
Open Circuit:   ON
Loss of Signal: ON

***** w1g1: E1 Tx Alarms *****

AIS:     OFF      | YEL:   ON


***** w1g1: E1 Performance Monitoring Counters *****

Line Code Violation     : 0
Far End Block Errors    : 0
CRC4 Errors             : 0
FAS Errors              : 0


Rx Level         : < -44db
```

1. First check the Rx Level
   The correct value is -2.5db
   Anything other than -2.5db indicates that there is a problem with the line.

   Options
   -2.5db - rx level is perfect
   -10db to -20db - there is something on the line but very weak. Could indicated a cable problem.
   -44db - there is nothing on the line. Either line is not started or there is no clock on the line.

   Sangoma cards will not come up if there is no clock on the line.
   One way to confirm that Telco is not giving us the clock, is to go back to TDM Physical Cofiguration section and configure the TDM Port for Master T1/E1 Clock. Note: Telco should

always supply the clock.

2. Rx Alarms

   Rx Alarms indicated that there is something wrong on the line
   RED - We are not receiving any kind of signal on the line.
   Usually indicates that the line is not active.
   AIS - The remote end is keeping us down on purpose
   Line in maintenance
   RAI - We receive good signal from remote end, but remote end does not see a good signal from us.
   Thus remote end is down.

3. Short/Open Circuit

   These statistics usually indicate cable issues.
   Or that the port is not plugged in at all.
   (Which in this example is the case)

For a description on what each alarm description and meaning, please see the following link: http://wiki.sangoma.com/Wanpipemon-T1-E1-physical-Line-alarms

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

21. Appendix

SS7 Overview

The *Common Channel Signaling System No. 7* (SS7 or C7) is a global standard that defines the procedures and protocols used to setup most of the world's public switched telephone network (PSTN) calls.The ITU definition of SS7 allows multiple national variants such as North America's American National Standards Institute (ANSI) and Europe's European Telecommunications Standards Institute (ETSI) standard.

Each time you place and release a telephone call that extends beyond the local exchange, SS7/C7 signaling takes place.

The SS7 network and protocol are used for:

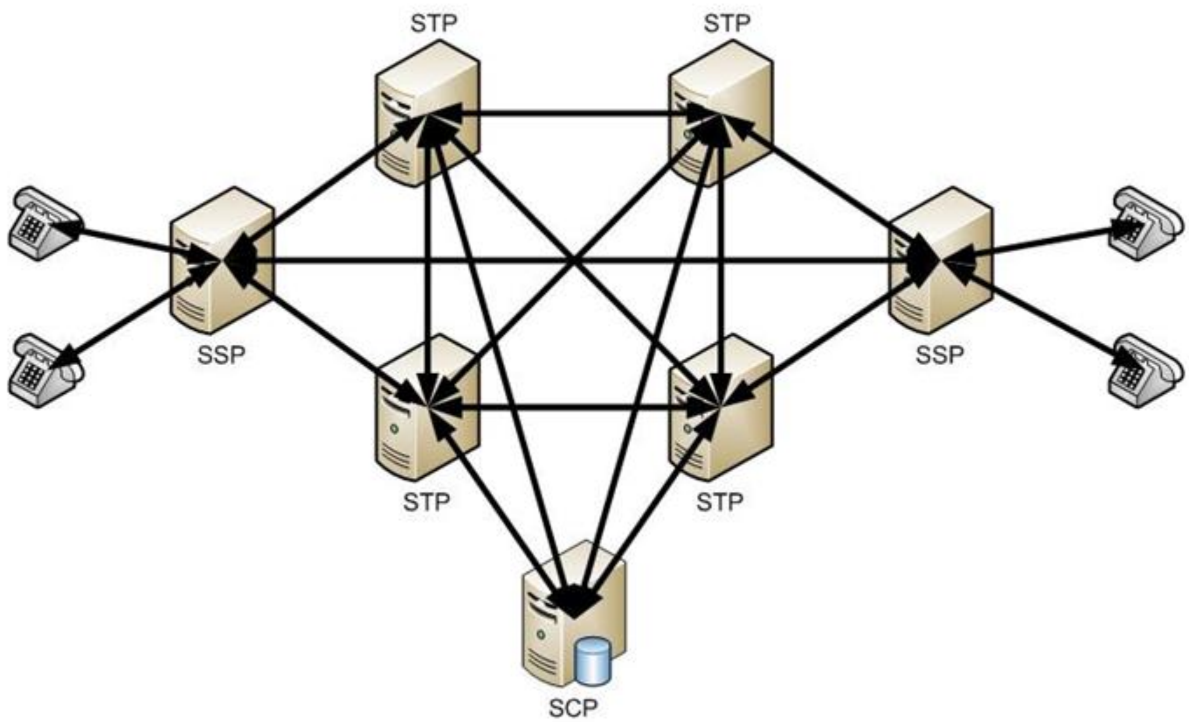
- To set up and tear down telephone calls
- Number translation (LNP)
- Toll-free (800) wireline services
- Wireless services such as SMS

SS7 messages are exchanged between two endpoints called signaling points on a 64 kpbs bi-directional channels (DS-0) known as signaling links. Each signaling point is uniquely identified by a numeric point code used to indentify the source and destination of each message.

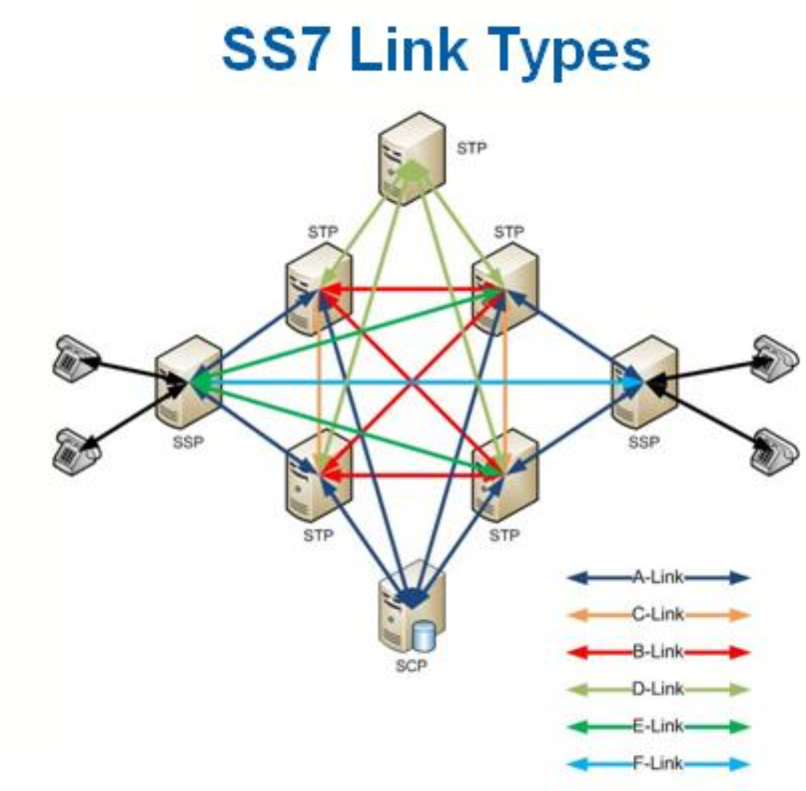
There are three types of signaling points in the SS7 network
:

- Service Switching Point (**SSP**)
  - Terminate signaling links
  - Start, end, and switch calls
- Service Transfer Point (**STP**)
  - Main routing switches
- Service Control Point (**SCP**)
  - Switches attached to databases

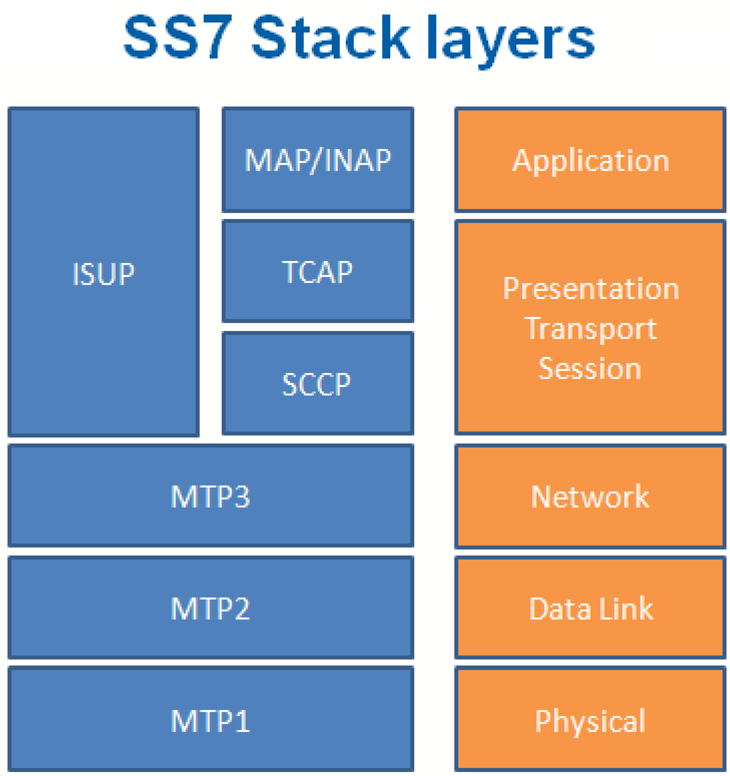
SS7 Signaling Points



Signaling links are categorized by link type ranging from A to F.



A Link (acces)	Link between an SSP or SCP to an STP. Its purpose is to deliver signaling messages
B Link (brigde)	Link between 2 mated STPs
C Link (cross)	Link between 2 STPs making them a mated pair
D Link (diagonal)	Link between 2 mated STPs (different hierarchical levels)
E Link (external)	Link between an SSP and a secondary mated STP
F Link (fully associated)	Link between 2 SSPs



The Message Transfer Part (MTP) consist of 3 layers

MTP1 is the physical layers protocol that can be E-1 (2048 kbps, 32 x 64kbps), DS-1 (1544kbps, 24 x 64kbps), V.35 (64kbps), DS-0 (64kbps) and DS-0A (56kbps).

MTP2 is the data link layer protocol that ensures reliable communications on a signaling link via error checking, flow control and sequence checking.

MTP3 is the network layer protocol that ensures reliable communications to other nodes in the network via addressing, routing and congestion control.

ISND User Part (ISUP) defines the protocols used to set-up, manage and release trunk circuits that carry voice and data on the PSTN.

Signaling Connection Control Part (SCCP) provides connectionless and connection-oriented network services and global title translation (GTT) capabilities above MTP level 3. SCCP is used as the transport layer for TCAP-based services.

Transaction Capabilities Applications Part (TCAP) supports the exchange of non-circuit related data between applications across the SS7 network using the SCCP connectionless service.

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223 sangoma.com

SIP Overview

SIP transactions and dialogs
A SIP signaling session between two User Agents (UAs) is composed of one or more SIP transactions. A SIP transaction occurs between a User Agent Client (UAC) and a User Agent Server (UAS). It might involve one or more intermediate SIP servers such as a proxy or redirect server. A SIP transaction comprises all messages that begin with the SIP request initiated from the UAC, until a final response (a non-1XX or non-provisional) is received from the UAS.

Typically, a SIP transaction comprises a SIP request message followed by one or more SIP response messages.

SIP messages

Each SIP transaction consists of a request that invokes a particular method, or function, on the server, and at least one response.

SIP requests

SIP requests are messages that are sent from client to server to invoke a SIP operation. RFC 3261 defines six requests or methods that enable a User Agent or SIP proxy to locate users and initiate, modify, and tear down sessions:

- INVITE: An INVITE method indicates that the recipient user or service is invited to participate in a session. This method can also be used to modify the characteristics of a previously established session.
- ACK: An ACK request confirms that the UAC has received the final response to an INVITE request. ACK is used only with INVITE requests.
- OPTIONS: An OPTIONS request is used to query servers about their capabilities. If the UAS is capable of delivering a session to a user, it responds with its capability set.
- BYE: A BYE request signifies the termination of a previously established session.
- CANCEL: A CANCEL request allows UACs and network servers to cancel an in-progress request, such as an INVITE.
- REGISTER: A REGISTER request is used to register the current contact information.

In addition, RFC 3515 defines the REFER method. This SIP extension requests that the recipient REFER to a resource provided in the request. This method can be used to enable many applications,

100 Renfrew Drive, Suite 100, Markham ON L3R 9R6 Canada • t. 1 905 474 1990 • f. 1 905 474 9223

including call transfer.

SIP responses

A server sends a SIP response to a client to indicate the status of a SIP request that the client previously sent to the server. Specifically, the UAS or proxy server generates SIP responses in response to a SIP request that the UAC initiates.
SIP responses are numbered from 100 to 600. For a complete list, see Section 21 of RFC 3261.

SIP message structure

A SIP message consists of the following four main components:

- start-line
- one or more header fields
- empty line indicating the end of header fields
- optional message body.

Start-line

The start-line for a SIP request is known as the Request-Line. The start-line for a SIP response is known as the Status-Line.
The Request-Line specifies the SIP method, the Request-URI, and the SIP version.
The Status-Line describes the SIP version, the SIP response code, and an optional reason phrase. The reason phrase is a textual description of the 3-digit SIP response code.

SIP headers

A SIP message is composed of header fields that convey the signaling and routing information for the SIP network entities (User Agent, proxy, B2BUA, and so on). Each header field consists of a field name followed by a colon (:) and the field value. For a description of the key SIP headers, refer to Section 7.3 of RFC 3261.

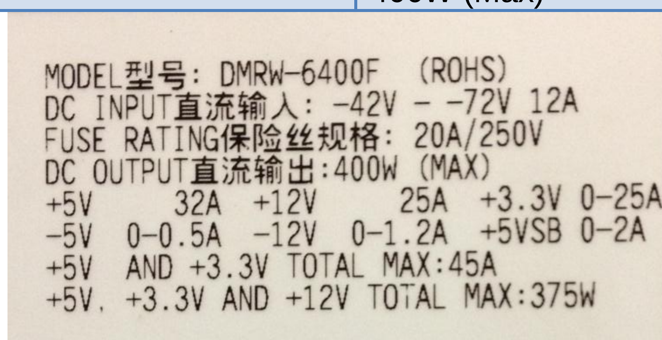
SDP body

The SDP (Session Description Protocol) body contains information about the message. The SDP body is optional. For a complete explanation of the SDP session description, see RFC 2327.

Redundant DC PSU

Sangoma NSG appliances come with redundant DC power supply.

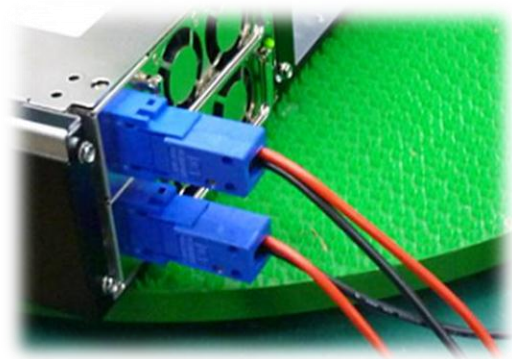
VOLTAGE	DC -36V ~ -72V
INPUT CURRENT:	12.0A (RMS). FOR -48 VDC
INRUSH CURRENT	20A (Max)
DC OUTPUT	400W (Max)



- **TEMPERATURE RANGE : OPERATING 100C --- 400C**
- **HUMIDITY:OPERATING:20%-95%, NON-OPERATING:10%-95%**
- **REMARKS:85% IS NORMAL CONDITION AND 95% IS WITH SPECIAL COATING PROCESS**
- **HOLD UP TIME: 1.6 ms MINIMUM AT FULL LOAD & NOMINAL INPUT VOLTAGE**
- **DIELECTRIC WITHSTAND: INPUT / OUTPUT 1500 VAC FOR 1 SECOND**
- **INPUT TO FRAME GROUND 1500 VAC FOR 1 SECOND**
- **EFFICIENCY: 65% TYPICAL, AT FULL LOAD**
- **POWER GOOD SIGNAL: ON DELAY 100 ms TO 500 ms, OFF DELAY 1 ms**
- **OVER LOAD PROTECTION: 130 ± 20%.**
- **OVER VOLTAGE PROTECTION: +5V → 5.5V ~ 7.0V, + 3.3V → 4.0V ~ 4.5V**
- **SHORT CIRCUIT:+5V,+12V,+3.3V**
- **EMI NOISE FILTER: FCC CLASS A, CISPR22 CLASS A**
- **SAFETY: UL 1950, CSA 22.2 NO/ 950, TÜV IEC 950**
- **REMOTE ON / OFF CONTROL**
- **THE UNIT SHALL ACCEPT A LOGIC OPEN COLLECTOR LEVEL WHICH WILL DISABLE / ENABLE ALL THE OUTPUT VOLTAGE (EXCLUDE +5V STANDBY), AS**

LOGIC LEVEL IS LOW, OUTPUTS VOLTAGE WERE ENABLE, AS LOGIC LEVEL IS HIGH, OUTPUTS VOLTAGE WERE DISABED • COOLING : TWO 40 mm DC FANS (MODULE) AC INLET IN EACH MODULE

DC PSU Cables



Connecting cables to a power supply depends on the remote power source.

Power Source Type	*Black Wire*	*Red Wire*
If power source -48V	-48V	0V (Ground)
If power source +48V	0V (Ground)	+48V

- The PSU **has** voltage reverse protection.
  If the red and black wires are connected the wrong way, the system will not power up. But there will be **no** damage to the PSU or the system.

Hot-swap procedures

Please refer to the following when either power module is defective.

- Locate the defective power module by examining the individual LED (if LED is distinguished, it indicates the power module is defective).

  *** WARNING
  please perform the following step carefully; otherwise, it may cause the whole system shutdown.
  *** WARNING

  Please do not remove the defective power module until you have worn gloves to keep from been burned. This is due to the cover of the power module is used as heat sink for cooling. Usually, its temperature is around 50-60 degree Celsius under full load condition.

- Loose the screws of power module bracket.
- Plug out the defective power module.

  *** WARNING
  please put aside the power module to wait for cooling down. Keep other people from toughing it until it is cooled.

- Replace a new / GOOD power module. Insert the power module into the power system till to the end.
- Check the LED of the power module, which should be in GREEN.
- Check the warning LED indicating the status of total power system, which should be in GREEN.
- Tighten the screws of the power module.
- If you want to test this new power module and simulate the defective situation, please refer to Section 1.7 Installation & Testing.

  Remarks: If the DC fan of the power module fails, you have to replace the power module. Please follow the Hot-Swap Procedures for replacement.

Trouble Shooting

If you have followed these instructions correctly, it should function normally.
Some common symptoms are, the system doesn't work, buzzer alarms, shutdown after running a very short period,… etc. If so, please check the following steps to verify and correct it.

- Check all connection (if pinouts is correct, if any connection loosed, if the direction is incorrect,… etc.).
- Check if any short-circuit or defective peripherals by plugging out the power connector from each peripheral, one at a time. Shall the system functions again, you have solved the problem.
- Once you hear the buzzer sound or see the warning LED in RED, please check,
- If the loading is under the minimum or over the maximum load of each channel.
- If the power source is well connected and supplied. Shall the above condition is happened, please disconnect the power source and wait for 2-3 minutes to release the protection status; then test it again.
- If buzzer keeps alarming or LED indicates the power module failure, please locate which power module is defective. Perform hot-swap procedures (ref. to Sec. 1.8 Hot-Swap Procedures). Return the defective power module back to your vendor for RMA procedure.
- If you cannot fix the problem, please contact your vendor for supporting.
  Note:
  * The description stated herein is subject to change without prior notice.
  * All brand names and trademarks are the property of their respective owners.