



Maidenhead Bridge



Cloud Security Connector GRE Cluster with Private Cloud Private Access

(For VMware, Hyper-V, KVM & Others)

Version 1.1

November 2021

Table of Contents

1 Introduction to Cloud Security Connectors for Netskope.....	6
2 Key benefits of the Cloud Security Connector GRE.....	6
3 Network Diagrams.....	8
3.1 Before (using Legacy SWG) and After (using CSC + Netskope).....	8
3.2 Information required to create the CSC.....	9
3.3 Example of Proxied Traffic to Netskope: ON/OFF Corporate Network.....	10
3.4 Example of Routed Traffic to Netskope: ON/OFF Corporate Network.....	11
3.5 Steering: Routing and Proxying all together.....	12
3.6 Private Cloud Private Access (PriCPA).....	13
4 Creating the Cloud Security Connector.....	14
4.1 Network Diagram - IP addressing.....	14
4.2 Adding your Public IP to the Netskope console.....	14
4.3 Filling the Form.....	15
4.4 What's next?.....	15
5 Firewall Requirements.....	16
5.1 NAT requirements.....	16
5.1.1 Checkpoint Firewall example.....	16
5.2 Allow Rules required.....	17
5.2.1 Outbound Rules:.....	17
5.2.2 Inbound Rules:.....	17
5.2.3 Checkpoint Firewall Example.....	17
6 Installing the OVA or Disk file in your Virtual Platform.....	18
6.1 Using VMware 5.x.....	18
6.2 Using VMware 6.x.....	19
6.3 Using Hyper-V.....	21
6.4 Using KVM.....	24
6.5 VM sizing.....	28
7 Powering up the CSC GRE.....	29
8 Steering traffic to NewEdge with the CSC GRE Cluster.....	31
8.1 Example of Proxied Traffic to Netskope: ON/OFF Corporate Network.....	31
8.1.1 Network diagram.....	31
8.1.2 Scenario, Objectives and Solution.....	31
8.1.3 Detailed configuration.....	32
8.1.3.1 Sites to Bypass.....	33
8.1.3.2 Configuring the Proxy Bypass on the CSC.....	33
8.1.3.3 PAC file for Company Devices: Laptops, desktops, servers, Etc.....	34
8.1.3.4 Netskope Client & Netskope Private Access.....	35
8.1.3.5 User experience On-Prem and Off-Prem.....	38
8.1.3.6 Final conclusion.....	39
8.2 Example of Routed Traffic to Netskope: ON/OFF Corporate Network.....	40
8.2.1 Network diagram.....	40
8.2.2 Scenario, Objectives and Solution.....	40

8.2.3 Detailed configuration.....	41
8.2.3.1 Sites to Bypass.....	41
8.2.3.2 Configuring the <i>Routed</i> Bypass Rules on the CSC.....	41
8.2.3.3 Netskope Client & Netskope Private Access.....	43
8.2.3.4 User experience On-Prem and Off-Prem.....	44
8.2.3.5 Final conclusion.....	46
8.3 Putting all together: Proxied and Routed Environments.....	47
8.4 Testing traffic to Netskope.....	48
8.4.1 www.netskope.com.....	48
8.4.2 https://ip.maidenheadbridge.com.....	49
8.4.3 SpeedTest.....	50
9 SSH Admin Console.....	51
9.1 Monitoring Tasks.....	52
9.1.1 Show Configuration and Status.....	52
9.1.1.1 General Information.....	53
9.1.1.2 Interfaces Information.....	53
9.1.1.3 TRAFFIC REDIRECTION Options.....	53
9.1.1.4 DNS INFORMATION.....	53
9.1.1.5 NETSKOPE INFORMATION.....	53
9.1.1.6 TUNNEL STATUS.....	54
9.1.1.7 HTTP://WWW.NOTSKOPE.COM PAGE STATUS.....	54
9.1.1.8 PROXY BYPASS - EGRESS INTERFACE STATUS.....	54
9.1.1.9 ROUTED BYPASS.....	54
9.1.1.10 AWS SSM AGENT.....	54
9.1.1.11 SYSLOG INFORMATION.....	54
9.1.1.12 HIGH AVAILABILITY Information.....	55
9.1.2 Show Interfaces Traffic.....	55
9.1.3 Traceroute and Latency Test.....	55
9.1.4 Speed Test.....	56
9.2 CSC Admin Tasks.....	56
9.2.1 AWS SSM Agent (Register / De-Register).....	56
9.2.1.1 Create the Key using "Hybrid Activations".....	56
9.2.1.2 Register the CSC on AWS.....	58
9.2.1.3 Checking the status of the AWS SSM agent.....	59
9.2.2 Manage Administrators.....	59
9.2.2.1 cscadmin settings.....	60
9.2.2.2 csccli user.....	60
9.2.3 Change Timezone.....	60
9.3 Proxy Bypass.....	61
9.3.1 Proxy Bypass - Traffic Flow.....	61
9.3.2 View Current Proxy Bypass List.....	61
9.3.3 Configure Proxy Bypass List.....	61
9.3.3.1 Auto - Proxy Bypass PAC URL.....	62
9.3.3.2 Manual.....	63

9.4 Routed Bypass.....	65
9.4.1 Routed Bypass - Traffic Flow.....	65
9.4.2 View Current Routed Bypass List.....	65
9.4.2.1 Compact.....	65
9.4.2.2 Json.....	66
9.4.3 Configure Routed Bypass List.....	67
9.4.3.1 Routed Bypass URL.....	67
9.4.3.2 Manual (Paste Routed Bypass JSON file).....	68
9.5 Log Information.....	69
9.5.1 View Current Month.....	69
9.5.2 View Last 6 Months.....	69
9.6 Configuration Wizards.....	70
9.6.1 Change Nodes, DNS servers, Syslog and more.....	70
9.6.1.1 Running the Configuration Wizard.....	71
9.6.2 Switch Tunnels - Primary / Secondary.....	74
9.6.3 Update Netskope Nodes Databases.....	74
10 Private Cloud Private Access.....	75
10.1 What is Private Cloud Private Access (PriCPA)?.....	75
10.2 PriCPA Network Diagrams.....	75
10.2.1 High Level Network Diagram.....	75
10.2.2 Low Level Network Diagram – PriCPA only.....	76
10.3 Configuring PriCPA.....	77
10.3.1 Create the Local configuration (first node of the cluster).....	77
10.3.2 Create the Local configuration (second node of the cluster).....	79
10.3.3 Create the Private Access Peers JSON file.....	80
10.3.3.1 Full mesh Private Access Peers JSON file.....	80
10.3.3.2 Understanding "privateApps" configuration and values.....	85
10.3.3.3 Example of "privateApps" for a Windows Domain controller.....	87
10.3.3.4 Example of "privateApps" for Internal Web Server.....	87
10.3.4 Load the "Private Access Peers JSON file" to the CSCs.....	88
10.3.4.1 Using "Private Access Peers URL".....	88
10.3.4.2 <i>Manual: Copy and Paste</i> "Private Access Peers Json file".....	93
10.4 Show Configurations and Status Private Access.....	94
10.4.1 Via SSH console.....	94
10.4.1.1 Show Peer/s Status.....	94
10.4.1.2 Show Peers Json file (active).....	95
10.4.1.3 Show Local Configuration.....	96
10.4.1.4 Show Firewall Local Rules.....	96
10.4.2 Via AWS Systems Manager or Rundeck.....	97
10.4.2.1 AWS Systems Manager.....	97
10.4.2.2 Rundeck.....	97
10.5 Configure CSC Remote Management via Private Access.....	98
11 Remote Management using AWS and Rundeck.....	99
11.1 AWS Systems Manager.....	99

11.1.1 Create Documents.....	99
11.1.2 Run Commands.....	101
11.1.3 List of Documents available for "Run Command".....	104
11.2 Rundeck.....	105
11.2.1 Jobs.....	106
11.2.2 Running job "Show Configuration and Status".....	106
12 DevOps operations.....	107
12.1 config.json file.....	108
12.2 routedBypassRulesFile.json.....	109
12.3 privateAccessPeersConfig.json.....	111
13 Appendixes.....	113
13.1 Appendix A: Routed Bypass JSON file if you don't have Cloud Firewall License.....	113
13.2 Appendix B: Release Notes.....	114
13.2.1 Version 1.0.....	114
13.2.2 Version 1.1.....	114
13.3 Appendix C: JSON formatters (Visual Code, Notepad ++).	115
13.3.1 Visual Code.....	115
13.3.2 Notepad ++.....	116
13.4 Appendix D: Securing an AWS Bucket by source IP.....	118

1 Introduction to Cloud Security Connectors for Netskope.

The Cloud Security Connector (CSC) is a device that enables easy deployments of the Netskope SASE solution in any customer environment.

The CSC's GRE Cluster lets you connect securely to Netskope NewEdge up to 1 Gbps without hassle.

The primary purpose of the CSC GRE family is simplicity: You don't need to re-architect your network. The CSC GRE is a direct replacement for your current Web Security Appliance. You can place the CSC GRE on the same network segment that your existing appliance and the CSC will redirect the traffic to Netskope NewEdge.

No configuration is required. Simply filling a form with your IP addressing, download the CSC (VM) and power it on.

The CSC GRE comes with all parameters to work with Netskope NewEdge. As soon you lunch the CSC at the location, the CSC will automatically connect to the best Netskope NewEdge nodes. The CSC GRE contains the perfect configuration for GRE tunnels, firewall rules and routing tables that are necessary.

You can run the CSC GRE on any virtual software: Vmware, Hyper-V, KVM, Etc, and a hardware version is also available on request.

All Netskope NewEdge functionalities are available. Internal IPs are completely visible on the Netskope NewEdge GUI.

Includes Private Cloud Private Access functionality that allows you to create a full mesh among the CSCs communicating your private traffic on a Zero Trust model.

Simple to install with complete management from Amazon AWS, Rundeck (or similar, like Ansible, Salt, Etc.) and SSH.

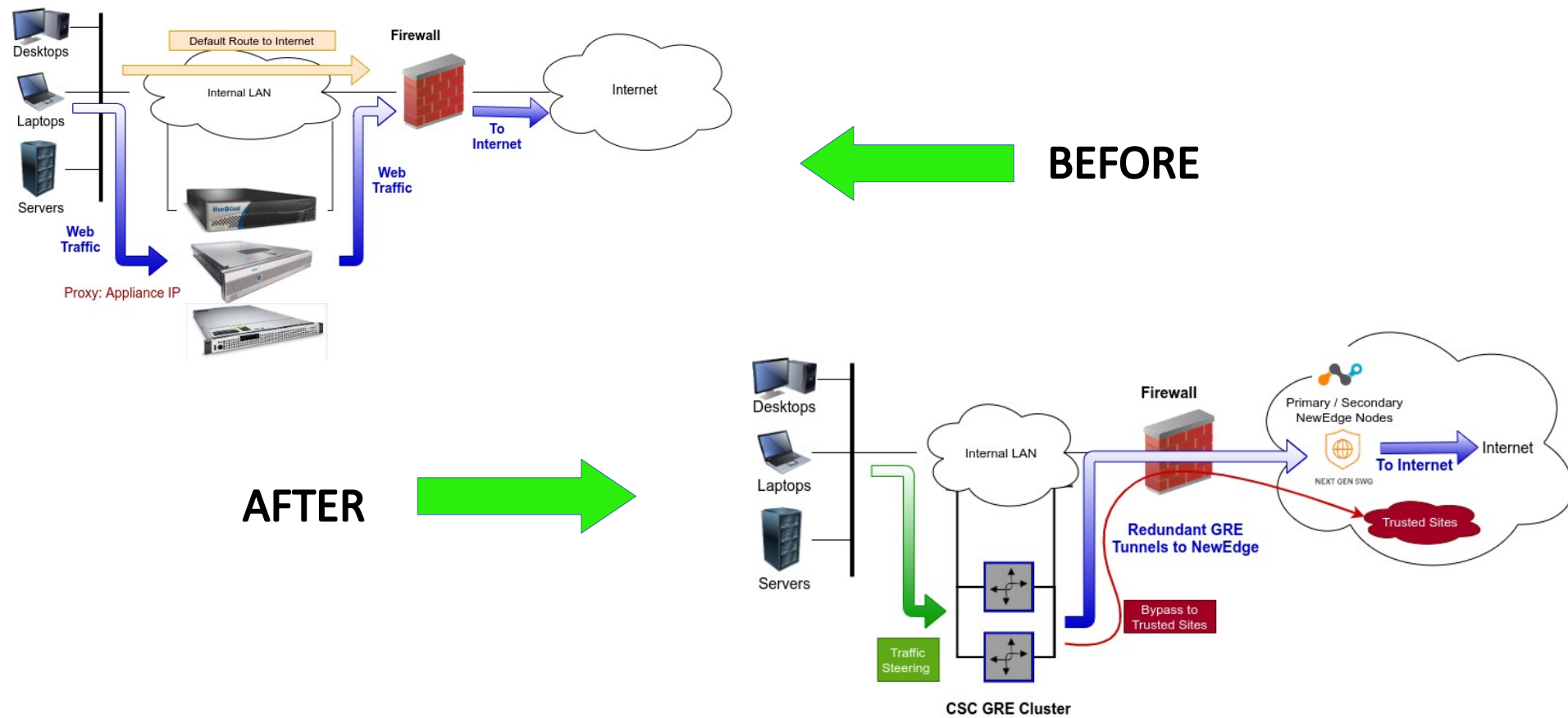
2 Key benefits of the Cloud Security Connector GRE

- No Networking knowledge is required.
- The CSC is a direct replacement for your current legacy Web Security Appliance.
- With Private Cloud Private Access you can connect all sites securely on a Zero Trust model. The CSC secures your Private Traffic between your locations.
- Enables any Location to be connected to Netskope NewEdge up to 1 Gbps.
- Easy to create: Filling a form indicating your IPs and GWs.

- Easy to deploy: Deploy OVA file setting the External and the Internal interface.
- The CSC comes with the optimal values to work with Netskope NewEdge and the best Nodes selected for your public IP.
- Full tunnel redundancy.
- High Availability.
- All traffic steering options supported:
 - Route all traffic to Netskope (or http/s only).
 - Use of PAC files.
 - Use of Explicit Proxy.
 - No default Route scenarios.
- Multiple options to Bypass Traffic:
 - Layer 7 Proxy Bypass to Trusted Web Sites.
 - Layer 4 Routed Bypass: TCP, UDP and ICMP per source/destination Network and Port (UDP/TCP)
- Cloud Firewall and Cloud Web Security.
- Complete visibility of internal IPs on Netskope Console.
- No operational burden for Administrators.
- Full hardened device.
- Works behind a NAT
- All virtual platforms supported: Vmware, Hyper-V, KVM, Etc. Hardware version available if required.
- Multiple tools for testing and troubleshooting included: Speed Test, MTR (MyTraceRoute), Keepalives statuses, Etc.
- Allow the internal communication between your locations with Private Cloud Private Access.
- Management via SSH, AWS Systems Manager, Rundeck or similar. (Ansible, Salt, Etc.)
- Small OVA instance: 2 CPU, 4 GB RAM, 16 GB Disk

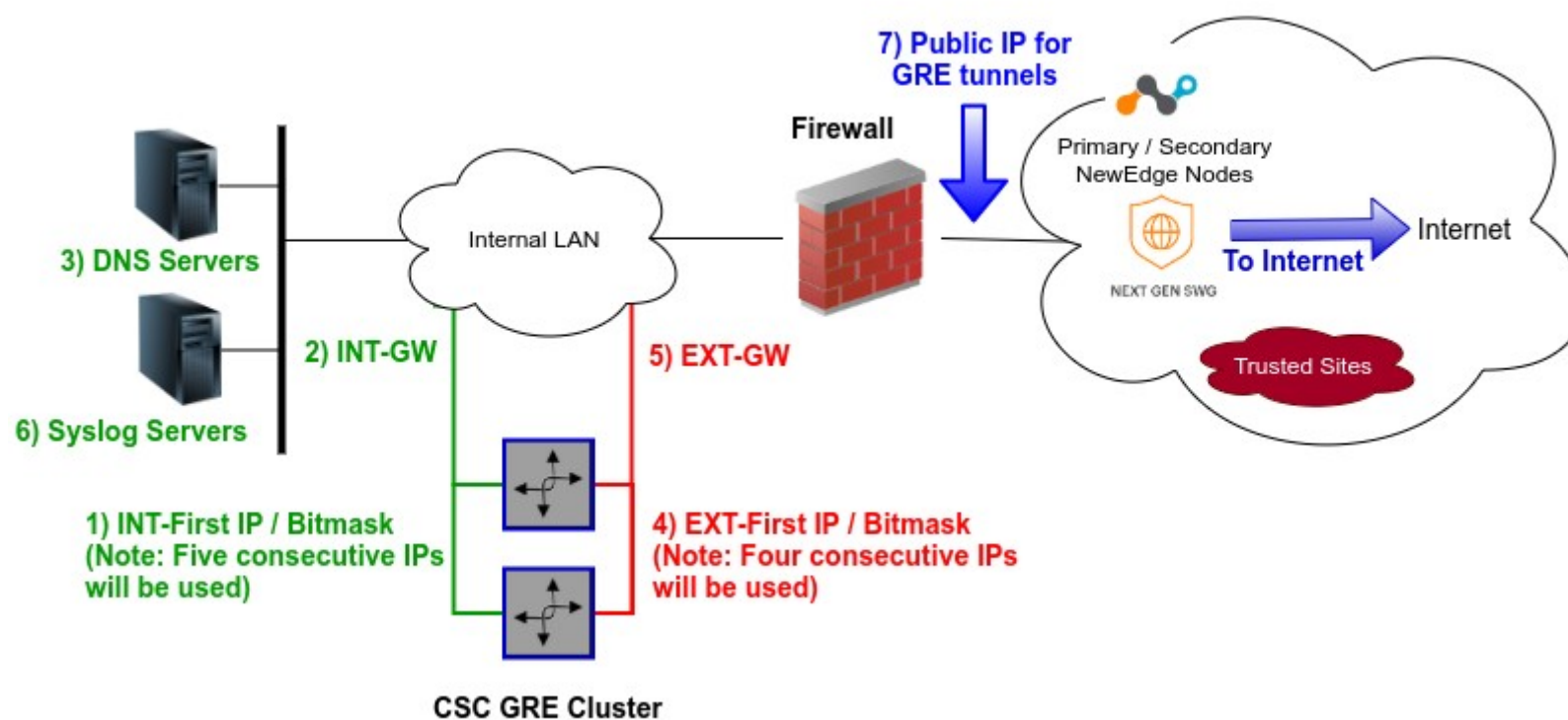
3 Network Diagrams

3.1 Before (using Legacy SWG) and After (using CSC + Netskope)



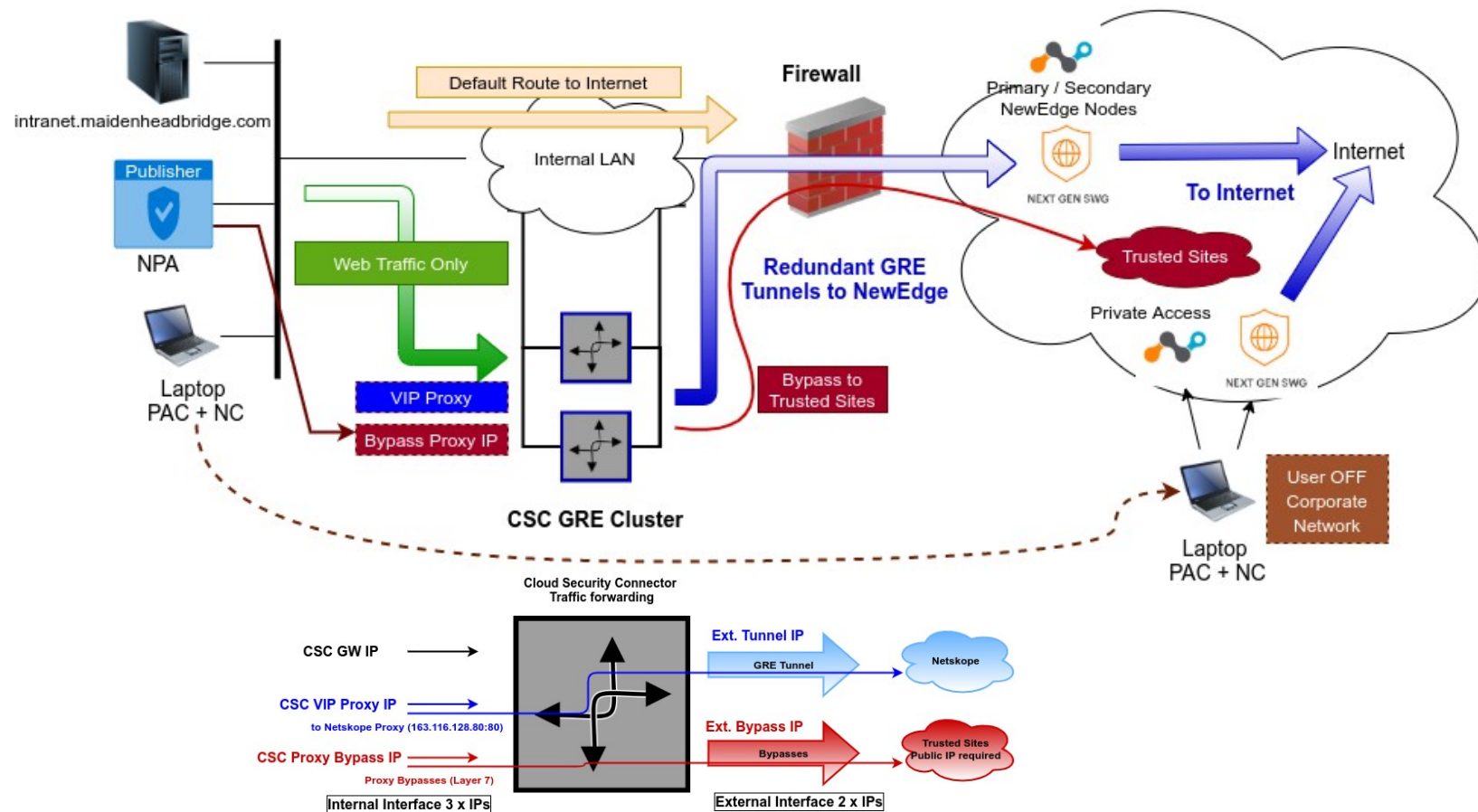
3.2 Information required to create the CSC

The following network diagram shows the information required to create the CSC GRE Cluster:



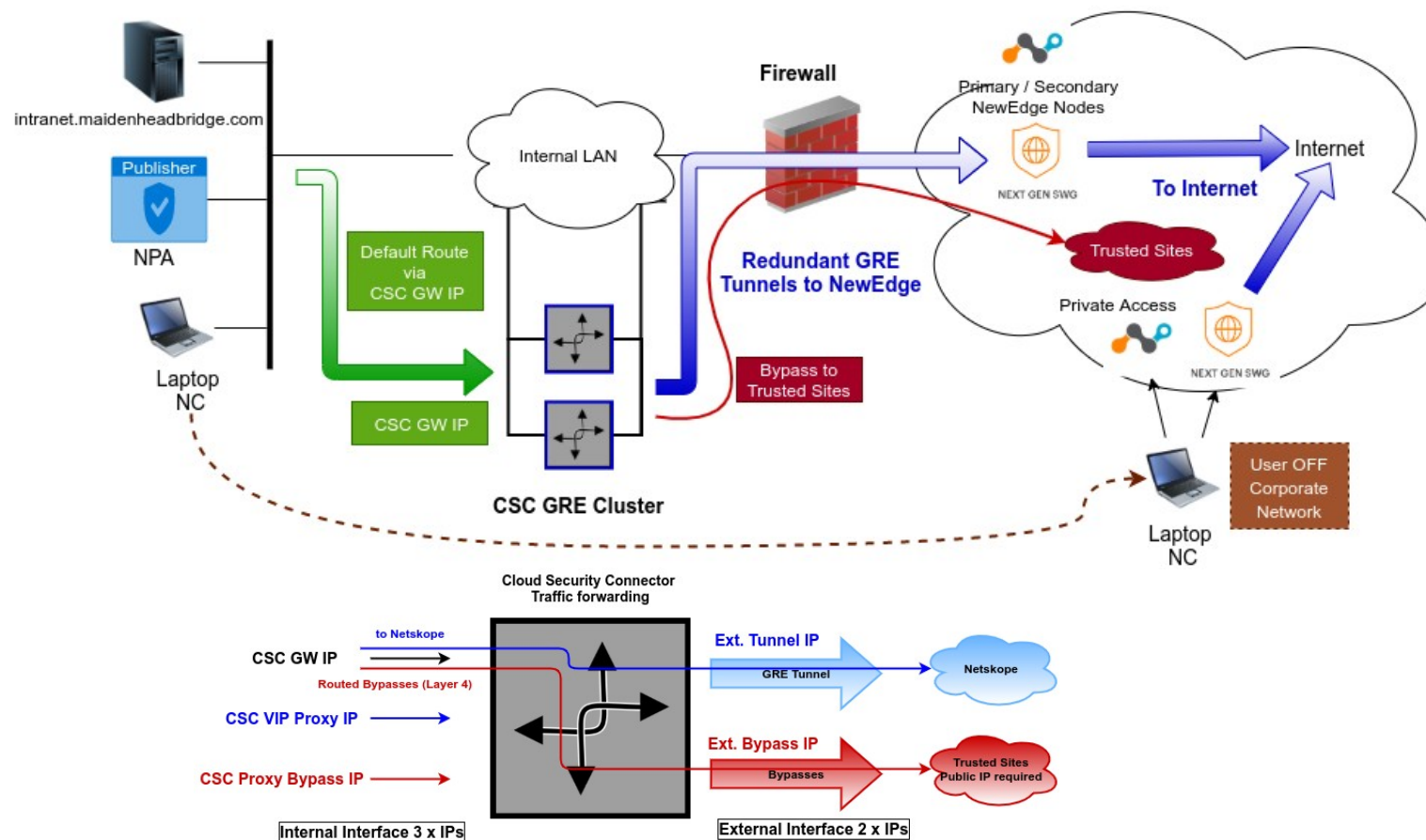
3.3 Example of Proxied Traffic to Netskope: ON/OFF Corporate Network.

This example shows how a User will have the same connectivity experience ON Corporate Network and OFF Corporate Network (at home, for example) thanks to the combination of the CSC + Netskope Web Security and Private Access.



3.4 Example of Routed Traffic to Netskope: ON/OFF Corporate Network.

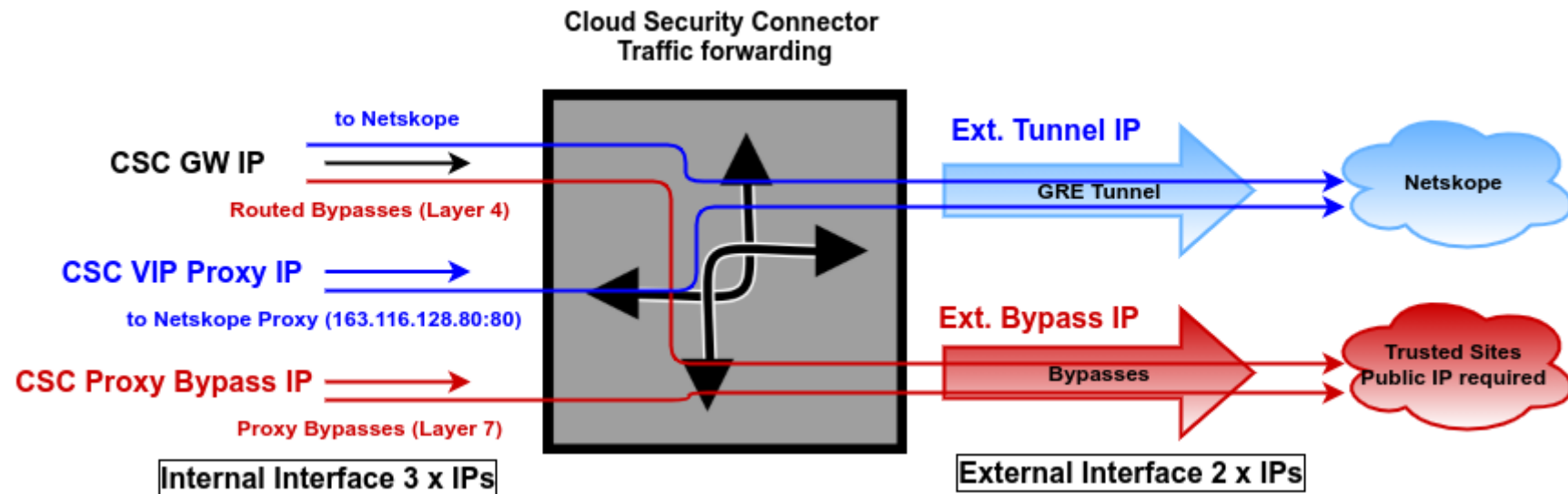
This example shows how a User will have the same connectivity experience ON Corporate Network and OFF Corporate Network (at home, for example) thanks to the combination of the CSC + Netskope Web Security and Private Access.



3.5 Steering: Routing and Proxying all together.

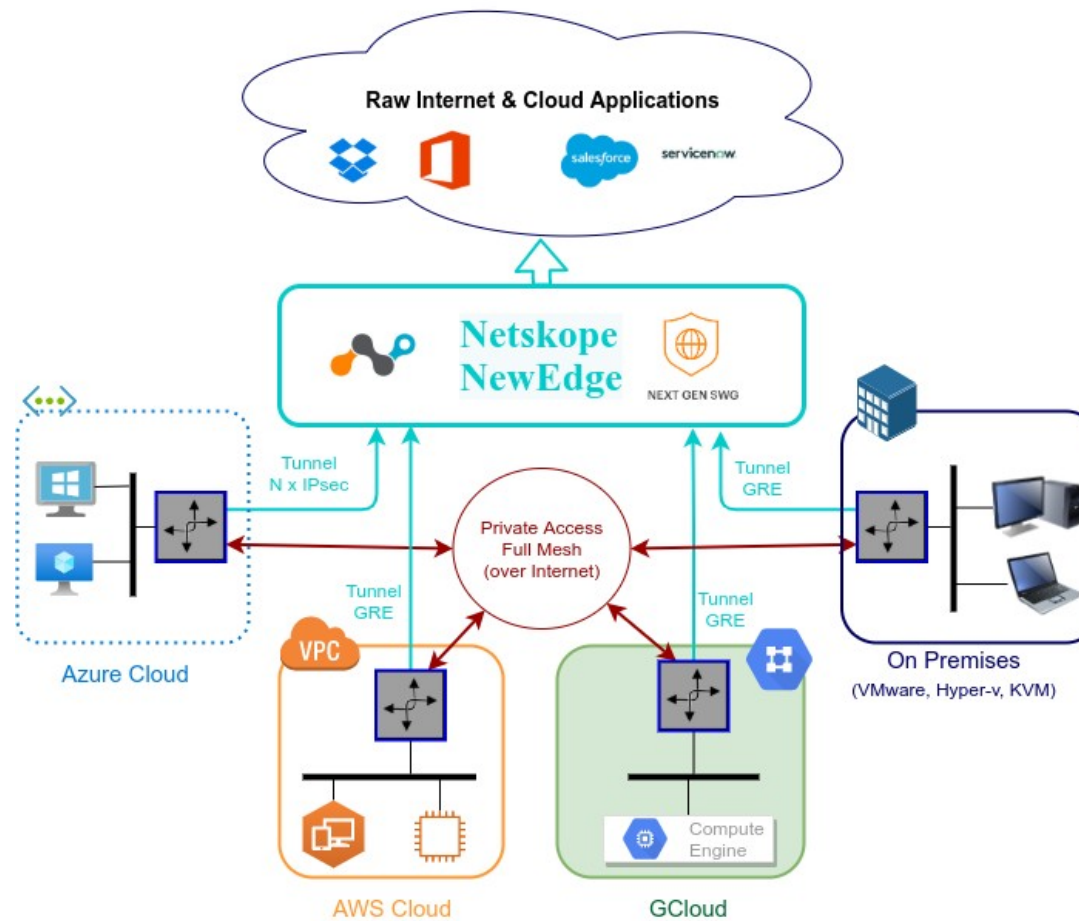
The most significant benefit of the Cloud Security Connector for Netskope is that it covers all possible scenarios (routed traffic, PAC files, explicit proxy, Etc.) for any device on your organization: Laptops, Desktops, Servers, IoT devices, Etc.

The following picture shows the CSC working with all scenarios combined.



3.6 Private Cloud Private Access (PriCPA)

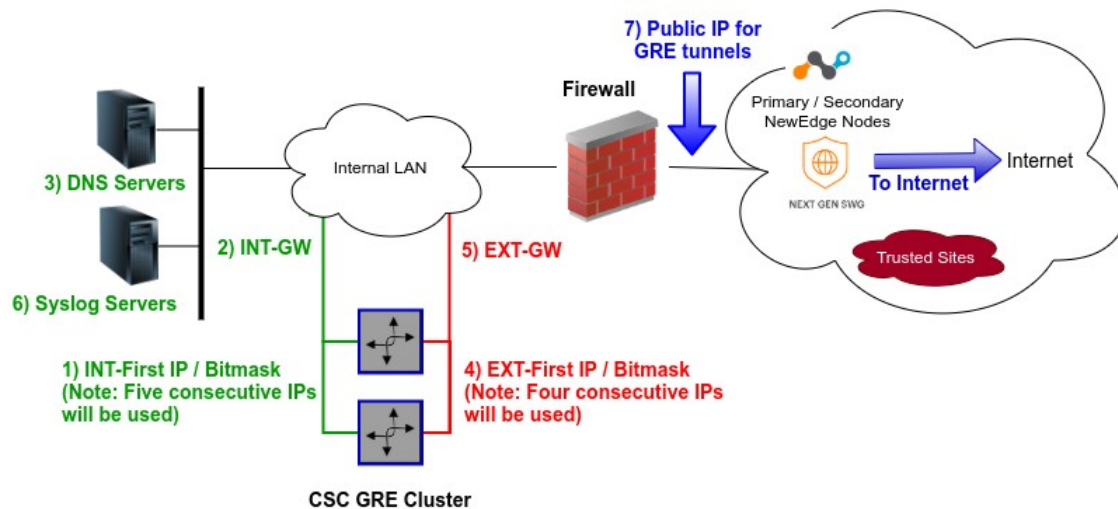
With the CSCs for Netskope, you can create your Private Cloud for connecting all your internal devices in a Zero Trust model from your physical and cloud locations.



4 Creating the Cloud Security Connector.

The creation of the CSC GRE Cluster is straightforward. Your only task is filling a form and to adding your Public IP on the Netskope console.

4.1 Network Diagram - IP addressing.



4.2 Adding your Public IP to the Netskope console

On your Netskope Console go to: Settings -> Security Cloud Platform -> (Traffic Steering) GRE and click "NEW GRE CONFIGURATION"

The screenshot shows the 'New GRE Configuration' form in the Netskope console. The form has three numbered red boxes: 1) CONFIGURATION NAME * with the value 'London-HQ'; 2) SOURCE PEER * with the value '200.200.200.200'; and 3) A button labeled 'SAVE'.

1. Put a Name for the GRE configuration in "Configuration Name"
2. Put your Public IP in "Source Peer"
3. Save.

You new configuration will appear on the GRE Menu.

GRE Configurations		
3 CREATED		
<input type="checkbox"/>	NAME ↕	SOURCE IDENTITY
<input type="checkbox"/>	London-HQ	200.200.200.200
		NETSKOPE POP
		None Connected

4.3 Filling the Form.

Before you start, you need to have this values ready:

- a) Five consecutive IPs for the internal interface of the CSC and its gateway IP.
- b) Four consecutive IPs for the external interface of the CSC and its gateway IP.
- c) (optional) DNS servers Primary and Secondary.
- d) (optional) Syslog/SIEM servers Primary and Secondary.
- e) The Public IP to be used for the GRE tunnels.
- f) (optional) Public SSH key/s for remote access to the CSC.
- g) (optional) Proxy Bypass: URL to PAC file.
- h) (optional) Routed Bypass: URL to JSON file.

The Form is at this URL: <https://bit.ly/2YG7SPm>

Please, follow the instructions on the Form.

4.4 What's next?

You will receive an email with the link to the URLs to download the OVA files of the Cloud Security Connectors.

5 Firewall Requirements

The CSC GRE Cluster uses four IPs on the external interface, and it is required to set up specific NAT and Allow Rules in your Firewall for all of them.

The following table shows the name and purpose of each one.

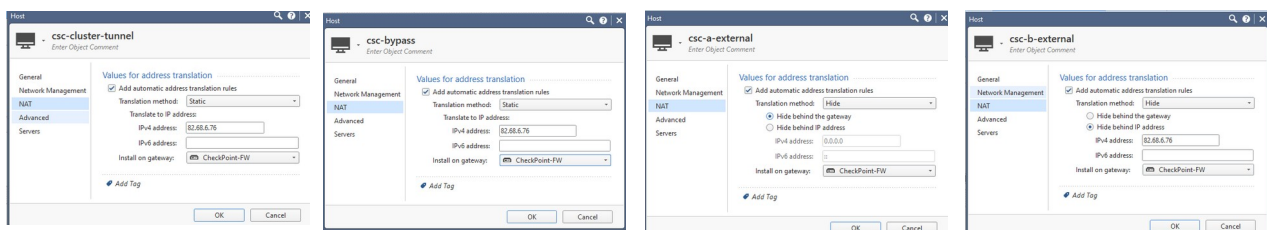
External IP#	Name	Purpose
First	GRE Tunnel IP	Source IP of the GRE Tunnel
Second	Bypass Egress IP	Source IP of the Routed, Proxied Bypasses and Private Access.
Third	CSC IP(eth0) -a	External IP of the VM -a
Fourth	CSC IP(eth0) -b	External IP of the VM -b

5.1 NAT requirements

External IP#	Name	NAT Type required:	via Public IP
First	GRE Tunnel IP	STATIC (also called 1:1 NAT) ¹	The GRE Public IP configured on the Netskope Console.
Second	Bypass Egress IP	STATIC ² or DYNAMIC.	Any Public IP - This will be your "Source Public IP" ³ when reaching Trusted Sites and connecting other Locations when using Private Access.
Third	CSC IP(eth0) -a	DYNAMIC (also called 1:N NAT)	Any Public IP
Fourth	CSC IP(eth0) -b	DYNAMIC (also called 1:N NAT)	Any Public IP

5.1.1 Checkpoint Firewall example

The images below shows the NAT on a Checkpoint Firewall.



- 1 Some firewall may require a dedicated IP when the protocol is GRE.
- 2 When using Private Access, it is required to use Static Nat to avoid changing the packet's Source Port.
- 3 Be sure that you are Natting the "Bypass Egress IP" via the Public IP configured on your "Trusted sites".

5.2 Allow Rules required

5.2.1 Outbound Rules:

The following table shows the allow rules required.

External IP#	Name	Protocol	Ports / Service	Destination
First	GRE Tunnel IP	GRE (47)	None. ⁽⁴⁾	Netskope Nodes
Second	Bypass Egress IP	TCP, UDP or ICMP	1 to 65535 ⁽⁵⁾	Internet, Trusted Destinations, Private Access Nodes.
Third	CSC IP(eth0) -a	PING	icmp_echo	Netskope Nodes and FW Gateway ⁽⁶⁾
Fourth	CSC IP(eth0) -b	TCP	80, 443	AWS Systems Manager ⁽⁷⁾ , Others ⁽⁸⁾

5.2.2 Inbound Rules:

External IP#	Name	Protocol	Ports / Service	Source
First	GRE Tunnel IP			
Second	Bypass Egress IP	UDP	Any (default 51820)	Private Access Nodes.
Third	CSC IP(eth0) -a			
Fourth	CSC IP(eth0) -b			

5.2.3 Checkpoint Firewall Example

No.	Name	Source	Destination	VPN	Services & Applications	Action
1	Allow Private Access Inbound	IP-217.155.196.81 IP-82.68.6.74	csc-bypass	* Any	UDP UDP-PORT-51820	Accept
2	Allow Private Access Outbound <i>Rules for Private Access</i>	csc-bypass	IP-217.155.196.81 IP-82.68.6.74	* Any	UDP UDP-PORT-51820	Accept
3	Proxy and Routed Bypass Traffic	csc-bypass	* Any	* Any	http https Teams-Skype-UDP	Accept
4	KeepAlives and AWS	csc-a-external csc-b-external	Netskope-London-LON1 Netskope-Manchester-MAN1 AWS-Destinations	* Any	https http icmp echo-request	Accept
5	GRE Tunnel	csc-cluster-tunnel	Netskope-London-LON1 Netskope-Manchester-MAN1	* Any	gre	Accept
6	Uplink Check - Allow Ping <i>Rules for Netskope</i>	csc-b-external csc-a-external	CheckPoint-FW	* Any	icmp echo-request	Accept

4 GRE is protocol and has not ports.

5 You can create rules on your FW enabling only what is on the bypass list or to allow "Bypass Egress IP" for all protocols and ports to any destination. Private Access traffic (UDP) will use this interface inbound and outbound.

6 The CSC GRE Cluster pings the Gateway IP of the Firewall to check reachability.

7 When using AWS SSM Agent, allow HTTPS from the csc-external-a (-b) to AWS. The AWS destinations are: ssm.<AWS region>.amazonaws.com, ec2messages.<AWS region>.amazonaws.com

8 The CSC retrieves the Proxy PAC URL, Routed JSON URL and the Private Access JSON URL via csc-external-a (-b). If you are hosting this URLs externally, please, create a rule to allow the CSC to reach them.

6 Installing the OVA or Disk file in your Virtual Platform.

The following examples shows the installation on Vmware and Hyper-V.

6.1 Using VMware 5.x

1. Go to vSphere, File > Deploy OVF template
2. Select the OVA File:

Source

[OVF Template Details](#)

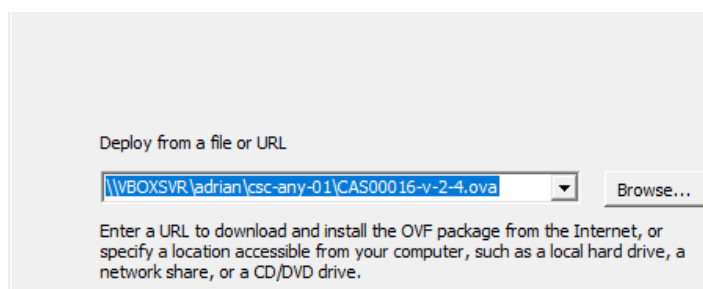
Name and Location

Resource Pool

Disk Format

Network Mapping

Ready to Complete



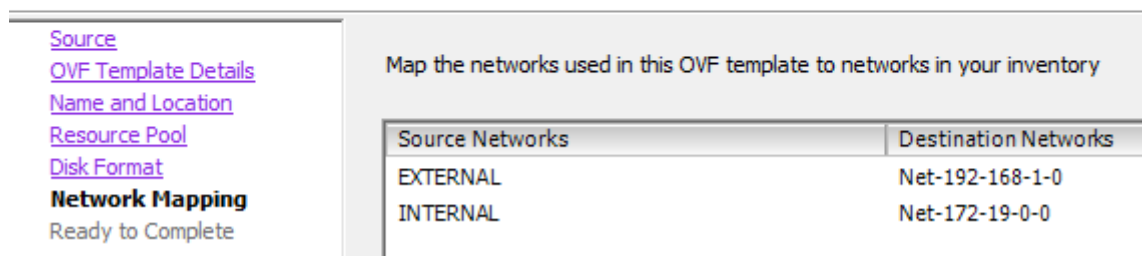
3. OVF Template Details: Click Next
4. Name and Location: Put the Name you want.
5. Resource Pool: Place the VM where you want.
6. Disk Format: Click Next
7. **Network Mapping: Please map the interfaces EXTERNAL and INTERNAL to your interfaces. Here an example:**



Deploy OVF Template

Network Mapping

What networks should the deployed template use?

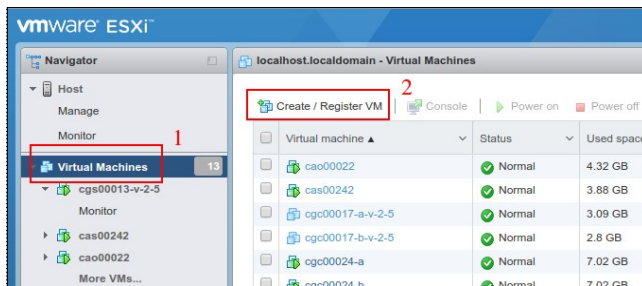


Source Networks	Destination Networks
EXTERNAL	Net-192-168-1-0
INTERNAL	Net-172-19-0-0

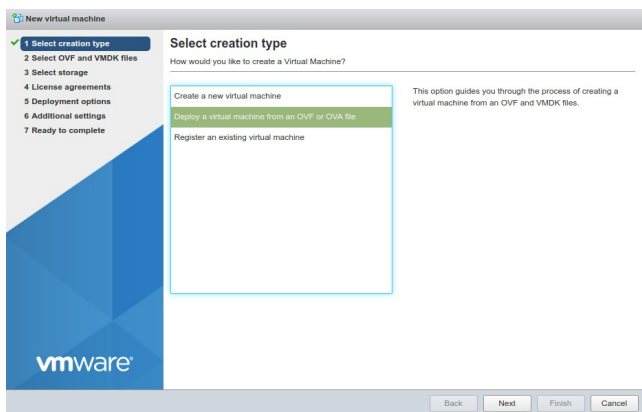
8. Click "Next"
9. Click "Finish"

6.2 Using VMware 6.x

1. Go to Virtual Machines → Create/Register VM

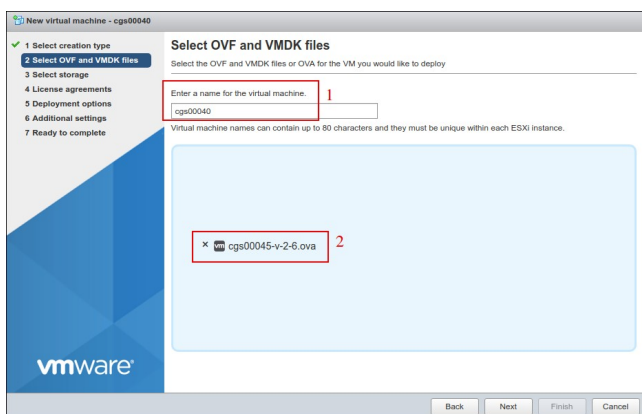


2. Deploy a virtual machine from an OVF or OVA file



3. Click "Next"

4. Put a "Name" and "Select the OVA File"



5. Click "Next"

6. Select Storage and click Next

7. On "Deployment options", Select:

- a) "Network Mappings" → Select "EXTERNAL" and "INTERNAL" interfaces of the CSC.
- b) Disk Provisioning: Thin
- c) Power on Automatically

The screenshot shows the 'New virtual machine - cgs00040' wizard. On the left, a progress bar indicates the steps: 1. Select creation type, 2. Select OVF and VMDK files, 3. Select storage, 4. Deployment options (current step), and 5. Ready to complete. The 'Deployment options' section has a sub-header 'Select deployment options'. It contains several fields: 'Network mappings' (labeled 1) with a dropdown menu showing 'EXTERNAL' (labeled 2) and 'INTERNAL' (labeled 3); 'Disk provisioning' (labeled 4) with radio buttons for 'Thin' (selected) and 'Thick'; and 'Power on automatically' (labeled 5) with a checked checkbox. At the bottom, there are 'Back', 'Next', 'Finish', and 'Cancel' buttons.

8. Click "Next"

9. The next screen will show all values:

The screenshot shows the 'Ready to complete' step of the 'New virtual machine - cgs00040' wizard. The progress bar on the left shows step 5 as the current step. The main area is titled 'Ready to complete' and 'Review your settings selection before finishing the wizard'. It contains a table with the following data:

Product	cgs00045
VM Name	cgs00040
Disks	cgs00045-v-2-6-disk1.vmdk
Datastore	datastore1
Provisioning type	Thin
Network mappings	EXTERNAL: Net-192-168-1-0, INTERNAL: Net-172-19-0-0
Guest OS Name	Unknown

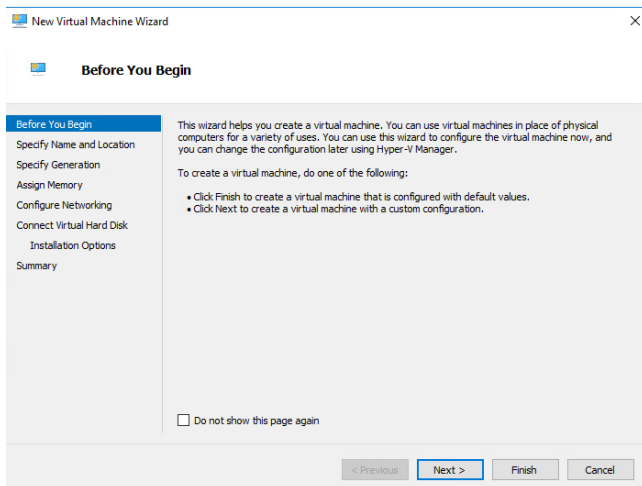
Below the table, there is a yellow warning icon and the text: 'Do not refresh your browser while this VM is being deployed.' At the bottom, there are 'Back', 'Next', 'Finish', and 'Cancel' buttons.

10. Click "Finish"

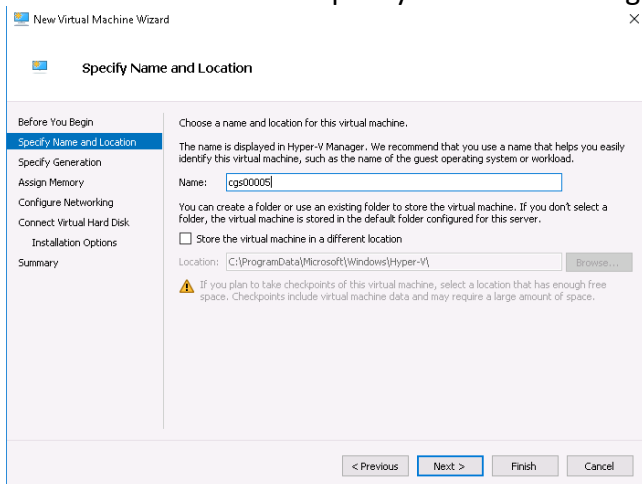
6.3 Using Hyper-V

Before to start: You will receive the CSC disk (.vhdx) on zip format. Please unzip it and place it on your Virtual Machine directory before to start this wizard.

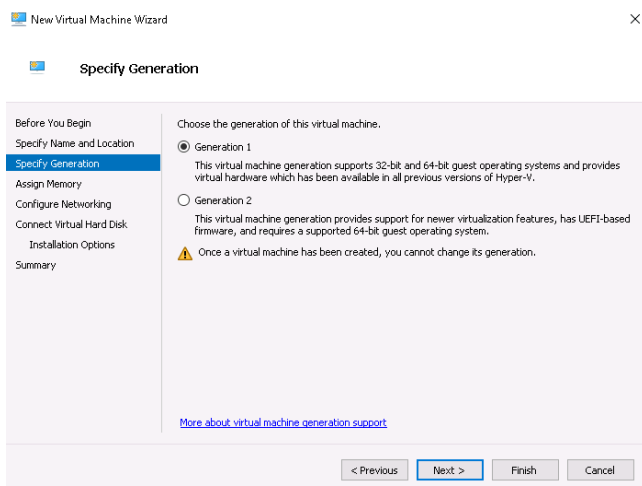
1. Go to Hyper-V and Click → Action → New



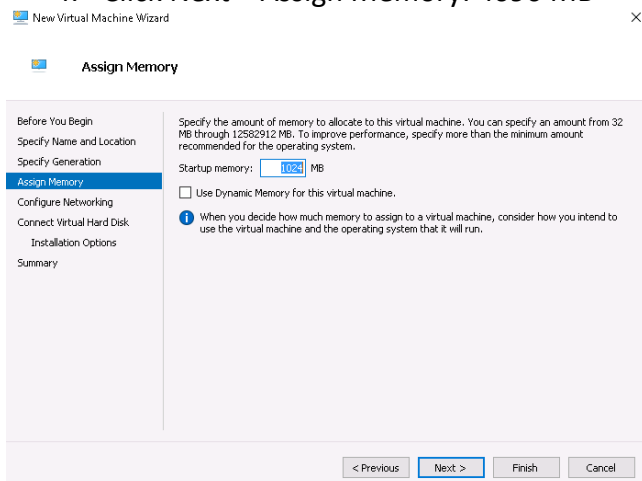
2. Click Next > and Specify Name and Storage



3. Click Next > Select "Generation 1"

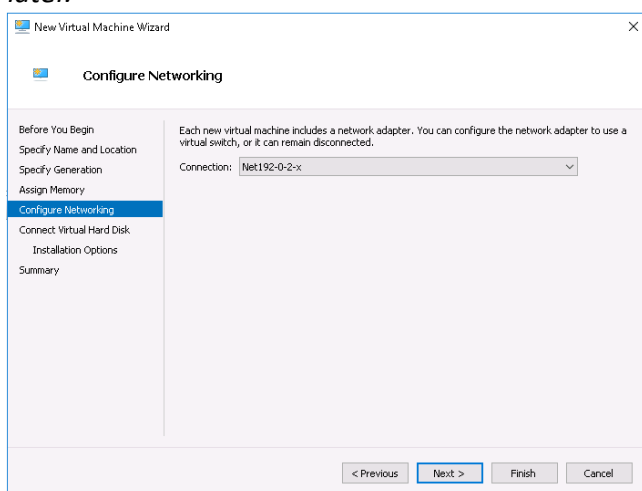


4. Click Next > Assign Memory: 4096 MB



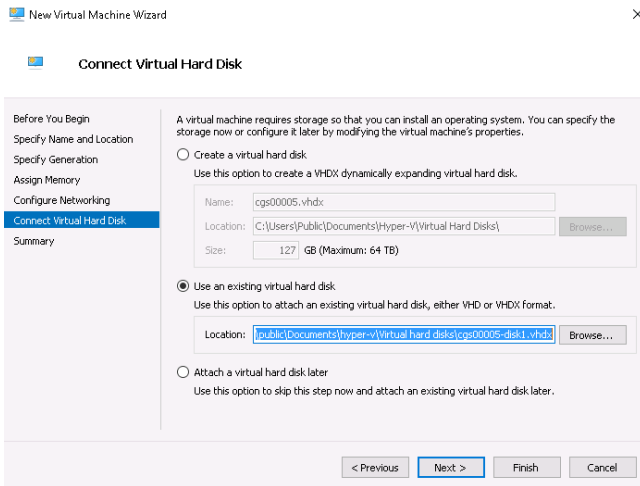
5. Click Next > Configure Networking

IMPORTANT: This is the EXTERNAL interface of the CSC. We are going to add the Internal Interface later.



6. Click Next > Connect Virtual Hard Disk

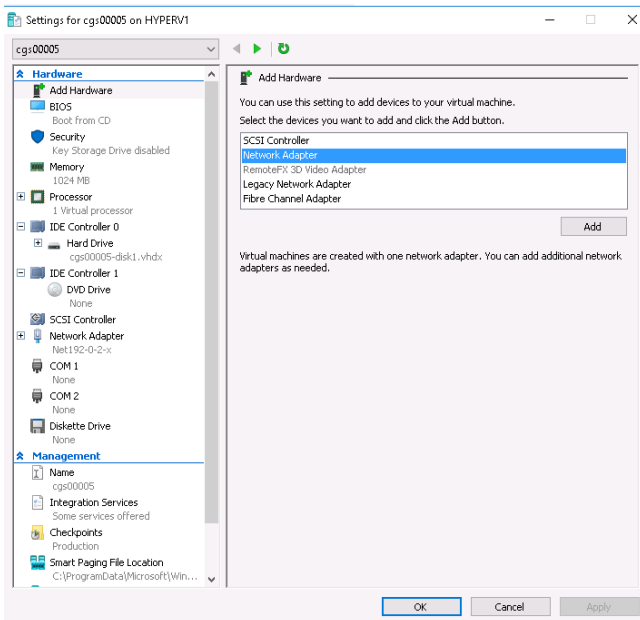
Select the unzipped disk on "Use an existing virtual disk"



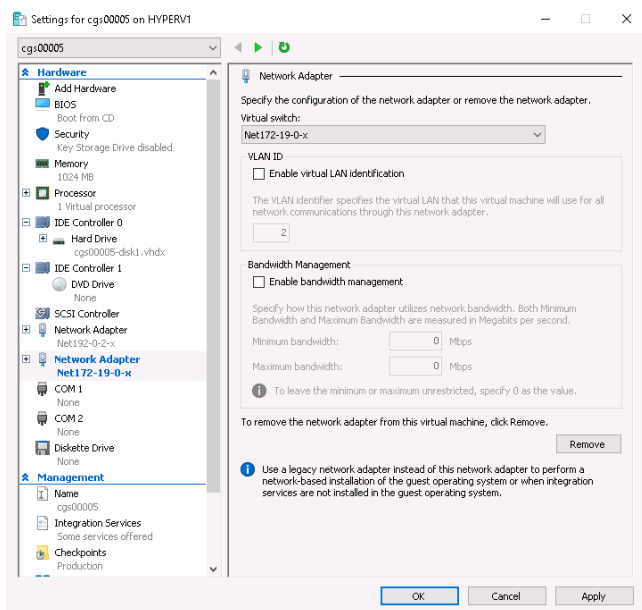
7. Click Next > Summary > Finish .

The machine will be created but we need to add the INTERNAL Interface.

8. Right Click the machine created > Settings > Add Hardware > Network Adapter



9. Click Add > and connect it to your INTERNAL virtual switch



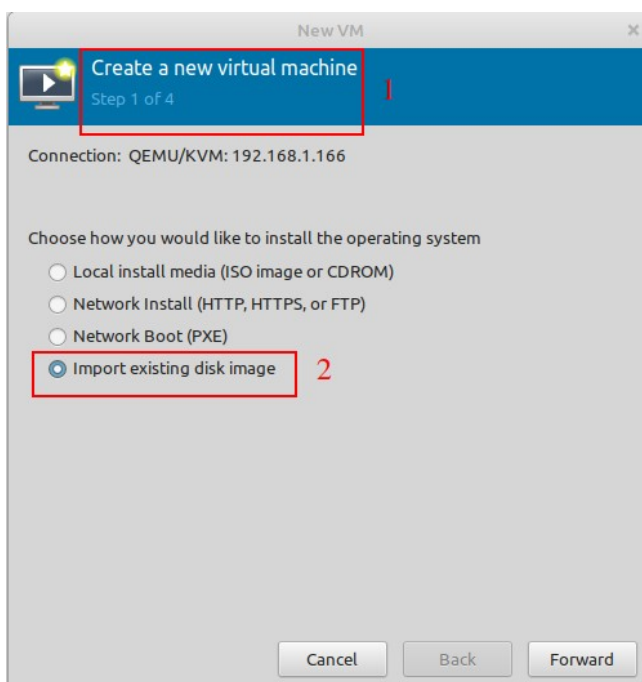
10. Click Apply and OK.

6.4 Using KVM

When using KVM, you will receive the disks of the CSC GRE Cluster in qcow2 format.

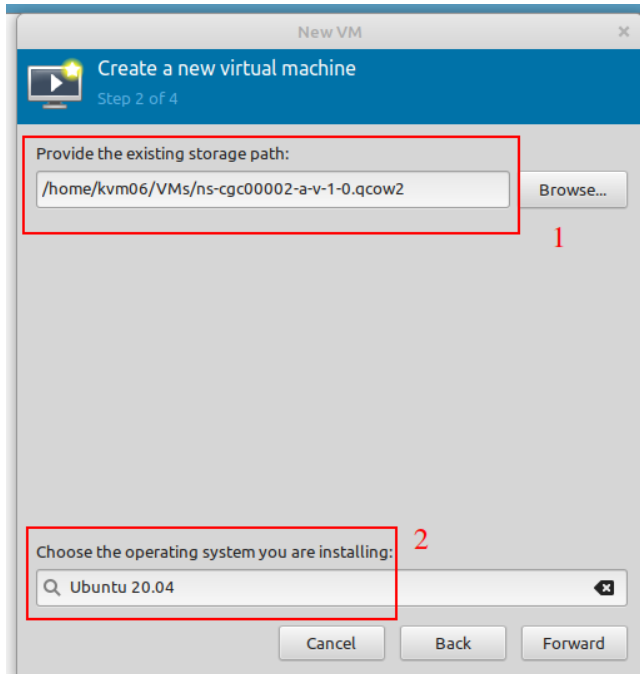
The following example shows the installation on a KVM server using Virtual Machine Manager (VMM)

1. Go to New -> Create a new Virtual Machine and select "Import existing disk image"



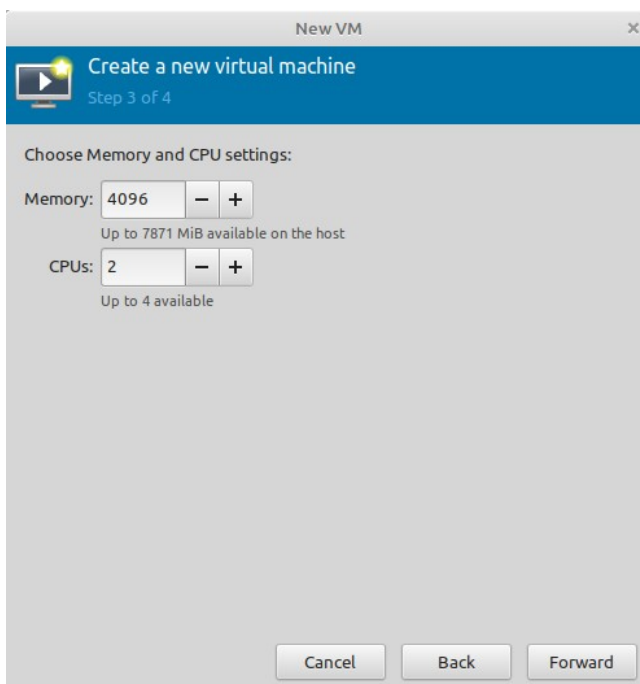
2. Click "Forward"

3. Browse for the Disk and select Ubuntu 20.04 (or another Ubuntu version if 20.04 is not available)



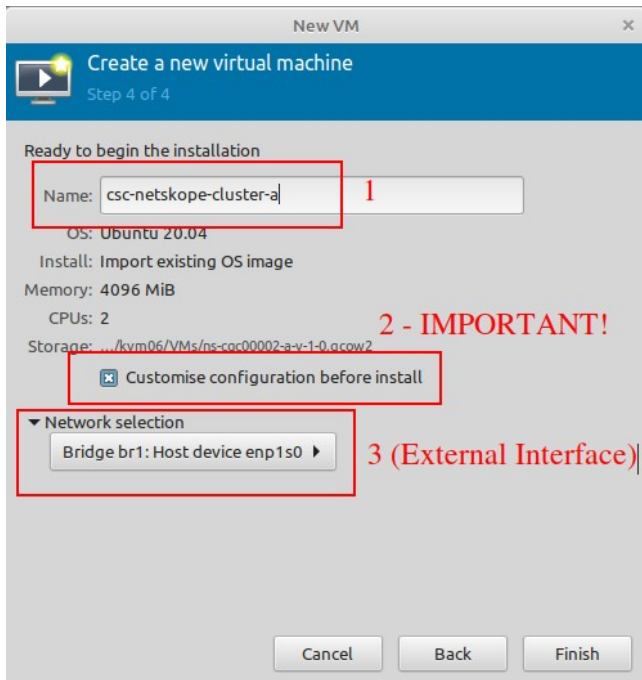
4. Click Forward.

5. Select 2 x CPU and 4 GB Memory (or more).



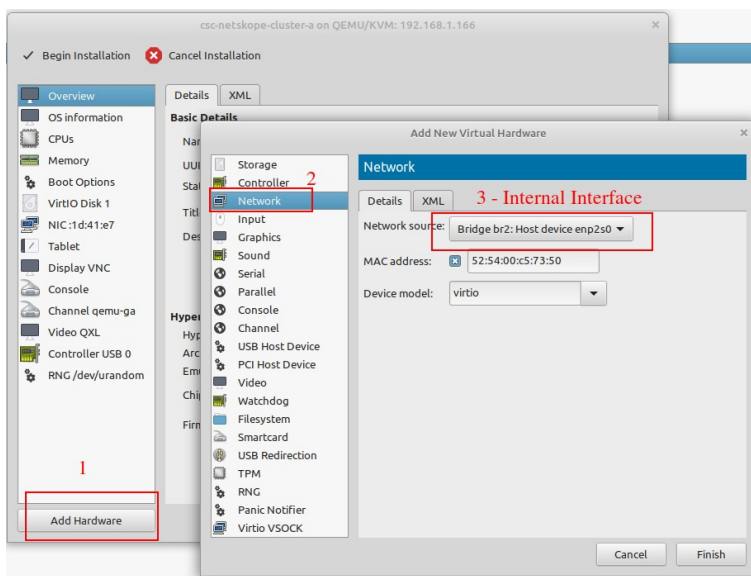
6. Click Forward.

7. Put the Name of the CSC, Select "Customise configuration before install" and choose here the External Interface.



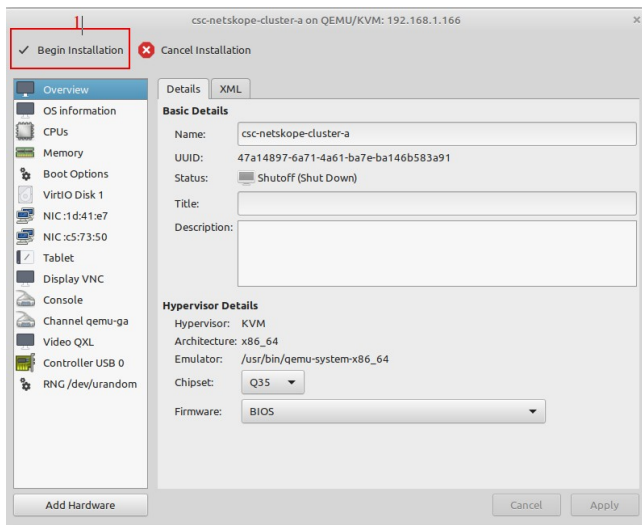
8. Click Finish.

9. We need to add now the Hardware for the Internal Interface. Click "Add Hardware", select "Network" and on Network Source choose the Internal Interface of the CSC.



10. Click Finish.

11. The last step is: "Begin Installation"



12. Done! Repeat the same process for the other CSC.

6.5 VM sizing

The CSC is a very efficient device and consumes few CPU and RAM resources. By default, we ship it with 2 x CPU, 4 x GB RAM and 16 GB disk. If you are going to have an intensive use of Proxy Bypass and high Private Access traffic, please increase CPU to 4 and RAM to 8 GB.

7 Powering up the CSC GRE

1. Power on the Virtual Machines.
2. SSH to the CSC using : `ssh cscadmin@< CSC IP(eth1) -a > or < CSC IP(eth1) -b >`. On the CSC GRE Cluster `CSC IP(eth1) -a` is the fourth internal IP and `CSC IP(eth1) -b` is the fifth.

When prompted, put the following username and password to login on the CSC Console:

Username: **cscadmin**

Password: **maidenheadbridge**

Note: SSH to the EXTERNAL interface IPs is not allowed.

```
Maidenhead Bridge
Cloud Security Connector GRE cluster for Netskope - Admin Console

Company : Maidenhead Bridge
Location : mhbdCp74
CSC ID : ns-cgc00004-a
Soft Version : 1.1

Please select an option by typing its number

Monitoring Tasks
1) Show Configuration and Status
2) Show Interfaces Traffic
3) Traceroute and Latency Test
4) Speed Test (Experimental)

CSC Admin tasks
5) AWS SSM Agent (Register or De-Register)
6) Manage Administrators
7) Change Timezone

Proxy Bypass
8) View Current Proxy Bypass List
9) Configure Proxy Bypass List

Routed Bypass
10) View Current Routed Bypass List
11) Configure Routed Bypass List

Log Information
12) View Current Month
13) View Last 6 Months

Configuration Wizards
14) Change Nodes, DNS servers, Syslog and more.
15) Switch Tunnels - Primary / Secondary.
16) Update Netskope Nodes Databases.

MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.

e) Exit

Selection: 
```

Select 1) Show Configuration and Status and Check Tunnel Status.

```
GENERAL INFORMATION
Company : Maidenhead Bridge
Location : HQkvm
CSC ID : ns-cgc00002-a
CSC date: Tue 12 Oct 09:27:00 BST 2021
Soft version : 1.0

INTERFACES INFORMATION
External: Tunnel IP: 192.168.1.60 | Bypass Proxy Egress IP: 192.168.1.61 | CSC IP(eth0): 192.168.1.62/24 | Network Gateway: 192.168.1.240 is Alive
Internal: CSC GW IP: 172.19.0.60 | CSC IP(eth1): 172.19.0.63/24 | Network Gateway: 172.19.0.133 is Alive

TRAFFIC REDIRECTION Options
To Netskope: VIP Proxy: 172.19.0.61:80 | Route all traffic via CSC GW IP | Netskope Global Proxy IP: 163.116.128.80:80 via CSC GW IP
Direct to Internet: Bypass Proxy: 172.19.0.62:3128 | Netskope Global Proxy IP: 163.116.128.80:3128 via CSC GW IP

DNS INFORMATION
DNS Server (1) IP: 172.19.0.100 is Alive
DNS Server (2) IP: 1.1.1.1 is Alive

NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74

Primary Tunnel:
    Node : GB,London,LON1
    Node Public IP: 163.116.162.36
    Node Probe: 10.162.6.209
Secondary Tunnel:
    Node : GB,Manchester,MAN1
    Node Public IP: 163.116.165.36
    Node Probe: 10.165.6.209

TUNNEL STATUS
Primary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
Secondary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
returnToPrimaryTunnel: true

Tunnel Status: Primary tunnel is active since: Fri 8 Oct 03:19:08 BST 2021

HTTP://WWW.NETSKOPE.COM PAGE STATUS
163.116.162.116 London, United Kingdom (LON1)

PROXY BYPASS - EGRESS INTERFACE STATUS
Proxy Bypass Egress Interface 192.168.1.61 can reach test page (https://ip.maidenheadbridge.com) via Public IP 82.68.6.73

ROUTED BYPASS
Using Routed Bypass URL: https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json
Routed Bypass URL https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json is reachable
Routed Bypass Rules configured via URL: 8

AWS SSM AGENT
AWS SSM Agent is active (running) since Tue 2021-10-05 20:23:47 BST; 6 days ago
Registration values: {"ManagedInstanceID":"mi-0160555d766bf22c6","Region":"eu-west-2"}

SYSLOG INFORMATION
SYSLOG Server (1) IP: 172.19.0.199 is Alive
SYSLOG Server (2) IP is not configured
SYSLOG TCP Port: 514

HIGH AVAILABILITY Information
This CSC (ns-cgc00002-a) is Cluster ACTIVE
```

Congratulations! You are connected to Netskope.

In the next chapter, we are going to discuss the multiple options to steer traffic to Netskope via the CSC GRE Cluster.

8 Steering traffic to NewEdge with the CSC GRE Cluster.

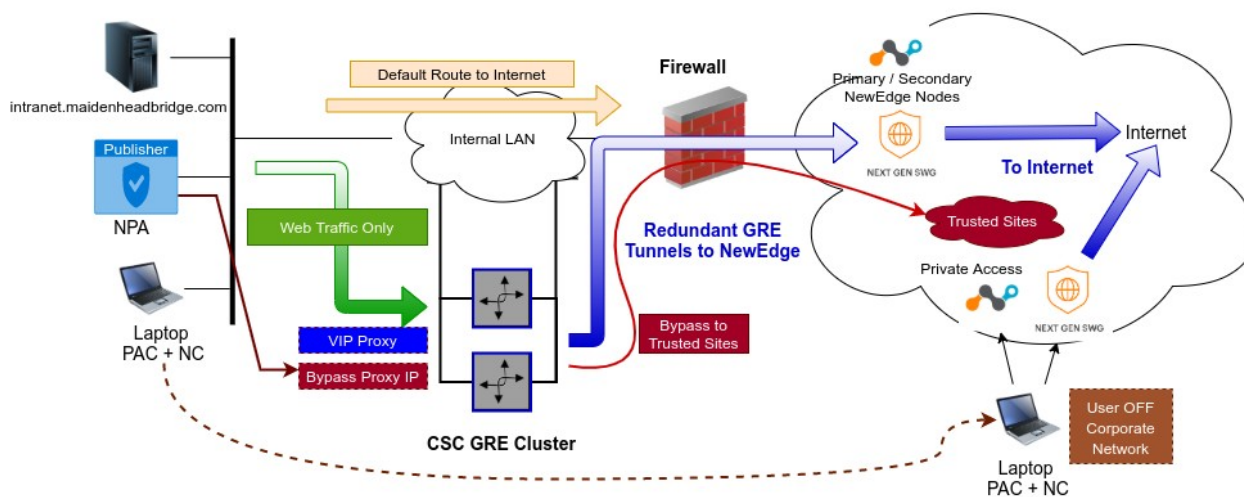
In Chapter 3 of this Administrator Guide, we showed the Network Diagrams of different scenarios of traffic steering. In this chapter, we are going to dig into more detail about the configuration required.

We are going to analyse three scenarios:

1. Using Proxy settings to steer traffic to Netskope: This scenario is for companies currently using PAC files (and maybe some client software) and migrating to Netskope Secure Web Gateway.
2. Routing all traffic to Netskope: This scenario is for companies who want to send all traffic to Netskope to use Cloud Firewall plus Secure Web Gateway.
3. Scenarios 1 and 2 combined: In this case, we will show how to use previous methods simultaneously and how the CSC helps to connect any device to Netskope NewEdge.

8.1 Example of Proxied Traffic to Netskope: ON/OFF Corporate Network.

8.1.1 Network diagram



8.1.2 Scenario, Objectives and Solution

Scenario

Before Netskope, the company used a legacy SWG appliance and now wants to migrate to Netskope Cloud SWG.

The default route to the Internet is via the Firewall.

The company devices used a mix of PAC files, Explicit Proxy Settings and software clients to redirect traffic to the legacy SWG.

When the user was OFF Corporate network, it required the use of VPN to access corporate sites or sites are reachable only if using the Public company IP.

Objectives

1. To protect Users when surfing the Internet.
2. To provide users with the same experience when they are ON or OFF Corporate Network.
3. To reach Microsoft O365 Authentication destinations using the company Public IP to apply Conditional Access rules and Multifactor Authentication controls.
4. To reach specific sites from the company Public IP.

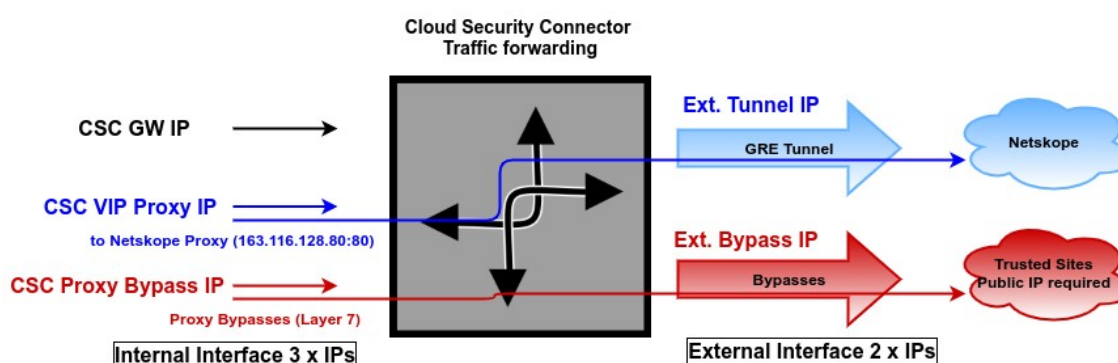
Solution

1. Deploy the CSC GRE Cluster in the same place as the Legacy SWG.
2. Create the PAC files to send traffic via the CSC GRE Cluster to Netskope and bypasses via the company Public IP.
3. Use the Netskope Client to steer traffic when the user is OFF the Corporate network and allow remote access to company applications.

8.1.3 Detailed configuration

In the proxy scenario, we will use the CSC VIP Proxy IP and CSC Proxy Bypass IP.

The flow of the traffic via the CSC will be the following:



CSC internal IP Name:	Proxy IP : PORT	Purpose
CSC VIP Proxy IP	172.19.0.61:80	Traffic to Netskope
CSC Proxy Bypass IP	172.19.0.62:3128	Traffic to Trusted Sites (Public IP required.)

8.1.3.1 Sites to Bypass

Sites required to be bypassed from Netskope in this example:

#	URL	Network / Subnet	Purpose
1	login.microsoftonline.com, login.microsoft.com, login.windows.net	20.190.128.0/18 40.126.0.0/18	O365 login URLs. Bypass required for Conditional Access Rules and MFA controls.
2	portquiz.net	52.47.209.216/32	Trusted site required to be reached via company Public IP.

8.1.3.2 Configuring the Proxy Bypass on the CSC

The recommended method to configure the Proxy Bypass is to create a PAC file and host the PAC file somewhere on the internet.

In this example, we are hosting the PAC file on an AWS bucket. Here are the details.

PAC URL	https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
PAC content	<pre>function FindProxyForURL(url, host) { // The value of bypassproxy here is not relevant. Leave this value as-is. var bypassproxy = "PROXY 172.16.1.1:3128"; // ===== // Section 3: bypassproxy via Cloud Security Connectors // bypassproxy via CSC Public IPs (Examples) // O365 Domains for ConditionalAccess if ((shExpMatch(host, "login.microsoftonline.com")) (shExpMatch(host, "login.microsoft.com")) (shExpMatch(host, "login.windows.net"))) // IP / Port test page (shExpMatch(host, "portquiz.net"))) { return bypassproxy } // ===== // This return sentence does nothing. It is for script compatibility purposes only. return bypassproxy }</pre>

The next step is to configure the PAC URL on all CSCs and refresh the Proxy Bypass list.

Later in this Guide, it is explained in more detail the configuration for Proxy Bypass.

8.1.3.3 PAC file for Company Devices: Laptops, desktops, servers, Etc.

Here the PAC file for company devices:

PAC URL	https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-laptop-desktop-pac-documentation.pac
PAC Content	<pre>function FindProxyForURL(url, host) { // ===== // Section 1: Standard PAC values var privateIP = /^(0 10 127 192\.168 172\.1[6789] 172\.2[0-9] 172\.3[01] 169\.254 192\.88\.99)\.[0-9.]+\\$/; var resolved_ip = dnsResolve(host); /* Don't send non-FQDN or private IP auths to us */ if (isPlainHostName(host) isInNet(resolved_ip, "192.0.2.0", "255.255.255.0") privateIP.test(resolved_ip)) return "DIRECT"; /* FTP goes directly */ if (url.substring(0, 4) == "ftp:") return "DIRECT"; // ===== // Section 2: Define Variables var tonetskope = "PROXY 172.19.0.61:80; DIRECT"; var bypassproxy = "PROXY 172.19.0.62:3128; DIRECT"; // ===== // Section 3: bypassproxy via Cloud Security Connectors // bypassproxy via CSC Public IPs (Examples) // O365 Domains for ConditionalAccess if ((shExpMatch(host, "login.microsoftonline.com")) (shExpMatch(host, "login.microsoft.com")) (shExpMatch(host, "login.windows.net"))) // IP / Port test page (shExpMatch(host, "portquiz.net"))) { return bypassproxy } // ===== // Section 4: Default Traffic // Default Traffic Forwarding. return tonetskope }</pre>

Sections of the PAC file for company devices:

Section 1: This section contains common values of a PAC file to send DIRECT traffic to internal networks (10/8, 172.16/12, 192.168/16, Etc.), FTP and others. You can leave this section as-is.

Section 2: In this section, you need to fill in the values of the variables tonetskope (via CSC VIP) and bypassproxy (via CSC Proxy Bypass IP)

Section 3: Section 3 is a copy/paste of the relevant information of PAC for the CSC and contains the list of destinations that will be bypassed using the company public IP.

Section 4: Default traffic will go via Netskope.

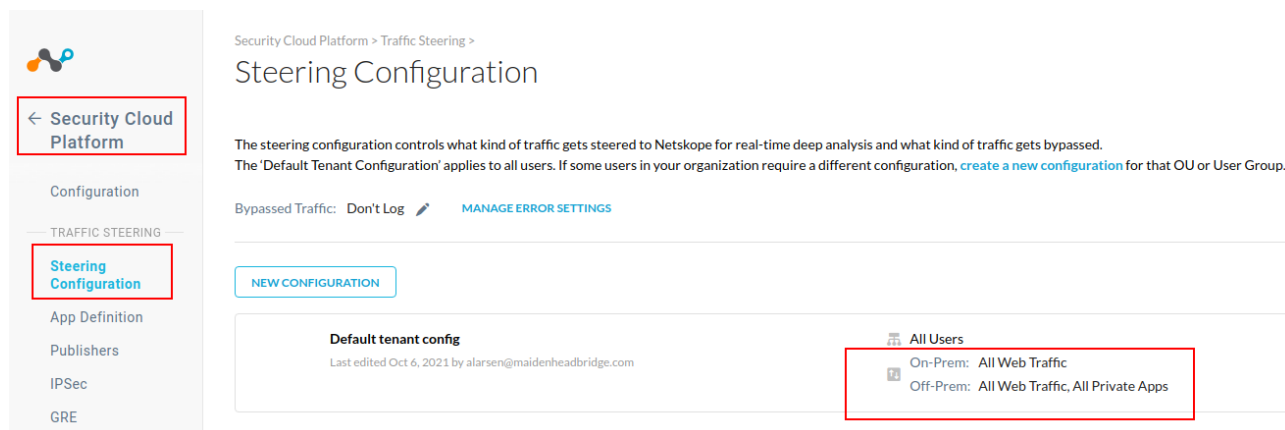
8.1.3.4 Netskope Client & Netskope Private Access

The last task is to install the Netskope Client and to configure three things:

1. Steering Configuration.
2. Detection ON/OFF Premises.
3. Create a Rule for Private Access to allow to reach the "bypassproxy" IP and Port when the user is OFF Corporate network. (OFF Premises)

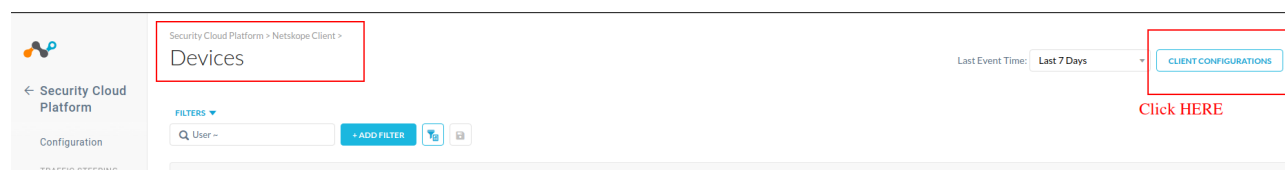
1. Steering Configuration

Go to Settings -> Security Cloud Platform -> Traffic Steering -> Steering Configuration and setup On-Prem and Off-Prem as shown below.



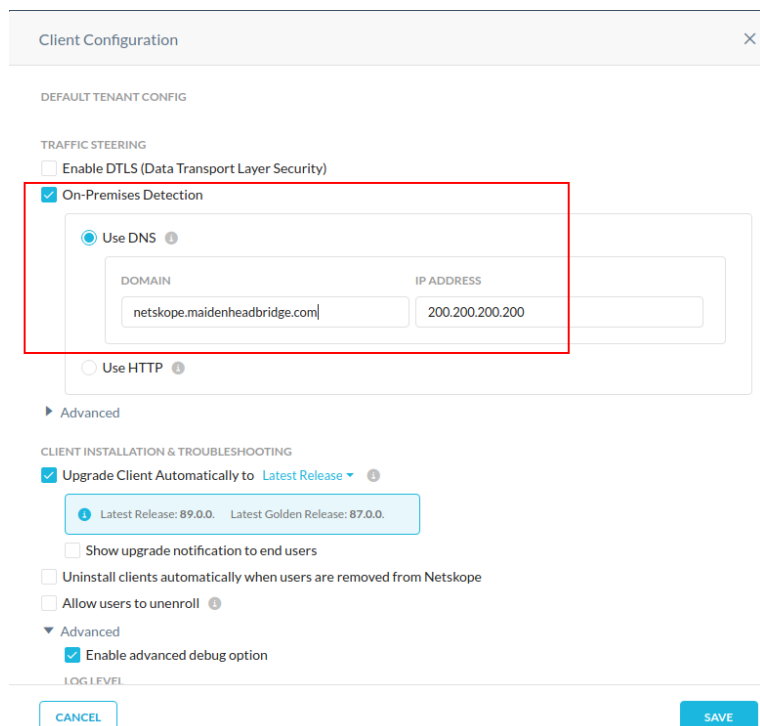
2. Detection ON/OFF Premises.

Go to Settings -> Security Cloud Platform > Netskope Client -> Devices, and click "Client Configurations" (right top corner)



Edit "Default Tenant Config" and enable "On-Premises Detection". In our case we created a DNS record A (netskope.maidenheadbridge.com) in our DNS server for this specific purpose.

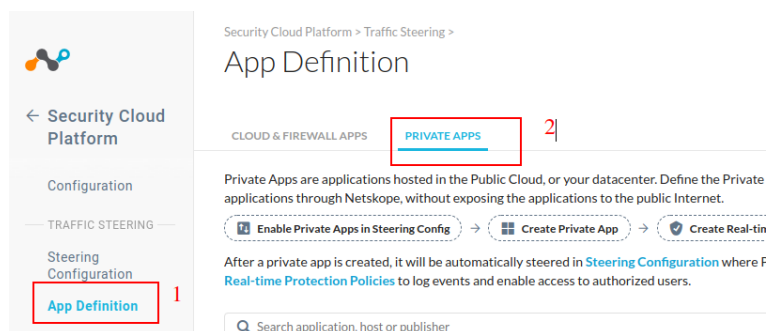
This DNS record is only available when the User is On-Premises.



The screenshot shows the 'Client Configuration' window with the 'DEFAULT TENANT CONFIG' tab selected. Under the 'TRAFFIC STEERING' section, the 'On-Premises Detection' checkbox is checked. Below it, the 'Use DNS' radio button is selected. The 'DOMAIN' field contains 'netskope.maidenheadbridge.com' and the 'IP ADDRESS' field contains '200.200.200.200'. The 'Use HTTP' radio button is unselected. Below this, the 'Advanced' section is expanded, showing 'Upgrade Client Automatically to Latest Release' checked, with a note 'Latest Release: 89.0.0. Latest Golden Release: 87.0.0.'. Other options like 'Show upgrade notification to end users', 'Uninstall clients automatically when users are removed from Netskope', and 'Allow users to unenroll' are unchecked. The 'LOG LEVEL' section is also visible at the bottom.

3. Create the App Definition and Rule for "bypassproxy" IP.

Go to Settings -> Security Cloud Platform > Traffic Steering -> App Definition and Select "Private Apps"



The screenshot shows the 'Security Cloud Platform' interface. The left sidebar has 'App Definition' highlighted under the 'TRAFFIC STEERING' section. The main content area is titled 'App Definition' and shows 'PRIVATE APPS' selected under 'CLOUD & FIREWALL APPS'. Below this, there are instructions on how to create private apps and a search bar for applications, hosts, or publishers.

and create the "Private App" for "bypassproxy" IP and Port (172.19.0.62 / 3128) in our example.

Edit Private App

Private apps are blocked by default. Policies are required to log events and enable access.

APPLICATION NAME

[CSC Proxy Bypass]

1

HOST

172.19.0.62

2

+ADD

PROTOCOL & PORT

TCP: 3128

3

UDP: Enter port or port range separated by commas (e.g. 443, 8080-8090)

PUBLISHER ⓘ

Publishers = MHB-HQ-Publisher

4

DNS RESOLUTION ⓘ

Minimum Publisher version of 1.4.6074 is required.

☐ Use Publisher DNS

CANCEL

SAVE

Finally, include the Private App on an Allow Rule on Policy Settings.

Go to Policies -> Real Time Protection -> New Policy -> Private App Access.

Policies

SSL Decryption
Real-time Protection
API Data Protection
Behavior Analytics
PROFILES
DLP
Threat Protection
Web
Connected App/Plugin
Domain
User

Real-time Protection Policy

Activities and actions available are dependent on the type of profile and applications you selected.

Source

User = [User]

ADD CRITERIA

Destination

Private App

Private App = [CSC Proxy Bypass]

ADD CRITERIA

Profile & Action

Action: Allow

ADD PROFILE

Set Policy

Allow PProxy Bypass

2

3

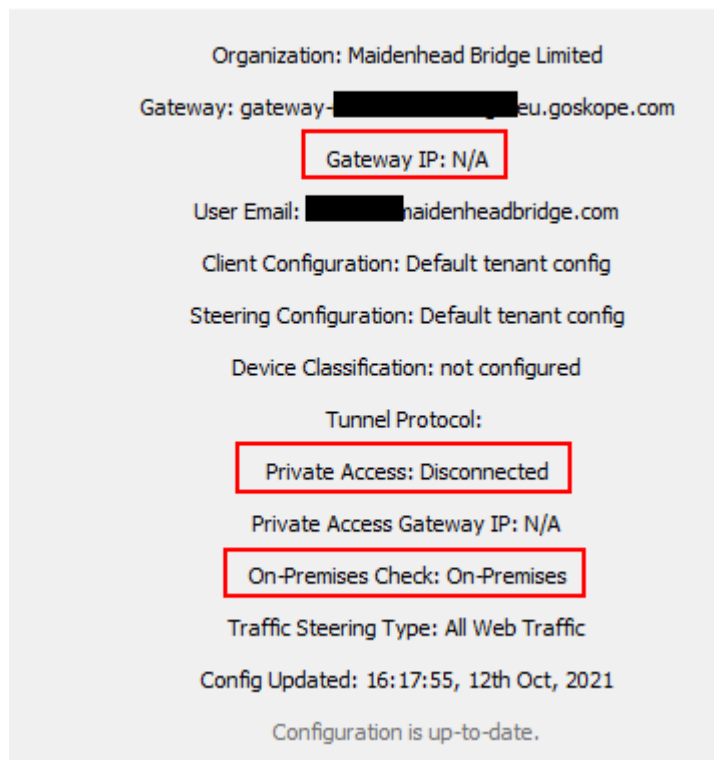
Done! Your Setup is complete!

8.1.3.5 *User experience On-Prem and Off-Prem*

On-Premises

When the User is On-Premises, the Netskope Client icon will show "Netskope Client disabled - GRE detected". Right Click the icon and click Configuration

Netskope Client Configuration

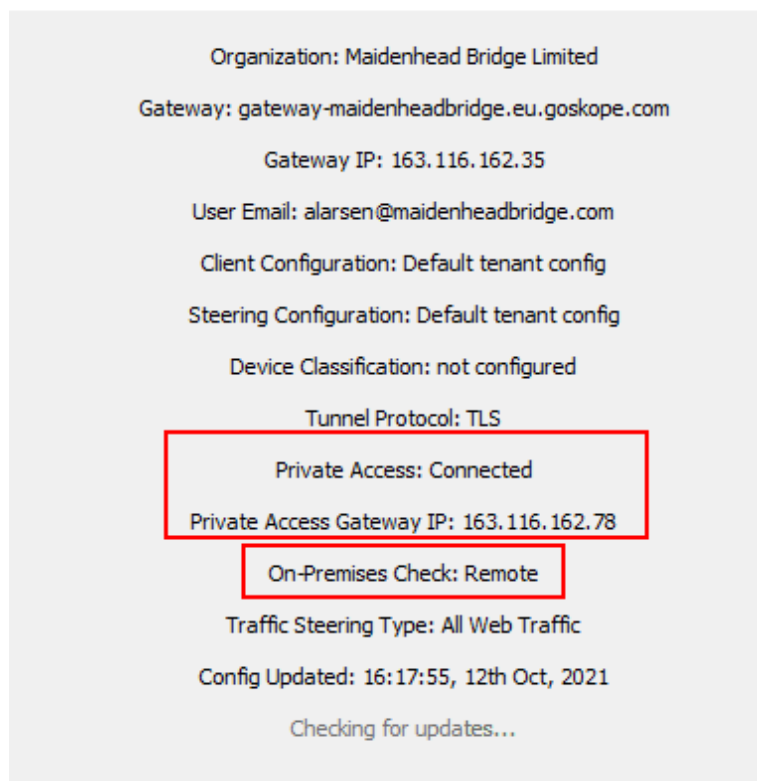


When On-Prem, the PAC file will command how to redirect the traffic to Netskope and Bypasses. Netskope will protect the user via the GRE tunnel.

Off-Premises

When the User is Off-Premises, the Netskope Client icon will show "Netskope enabled". Right Click the icon and click Configuration:

Netskope Client Configuration



In this case, the user traffic to the Internet is protected using the Netskope client tunnel, and Private Access is connected, allowing the bypasses to go via the company public IP. How? Below is the explanation.

1. The user types "portquiz.net" on the Browser.
2. The Browser reads the PAC file and uses Proxy 172.19.0.62:3128 to reach "portquiz.net"
3. Private Access intercepts the communication to "Private App" -> 172.19.0.62:3128 and sends the traffic to the Publisher that is on the company behind the CSC.
4. Finally, the flow reaches the CSC Proxy Bypass IP and leaves to the Internet via the Corporate Public IP.

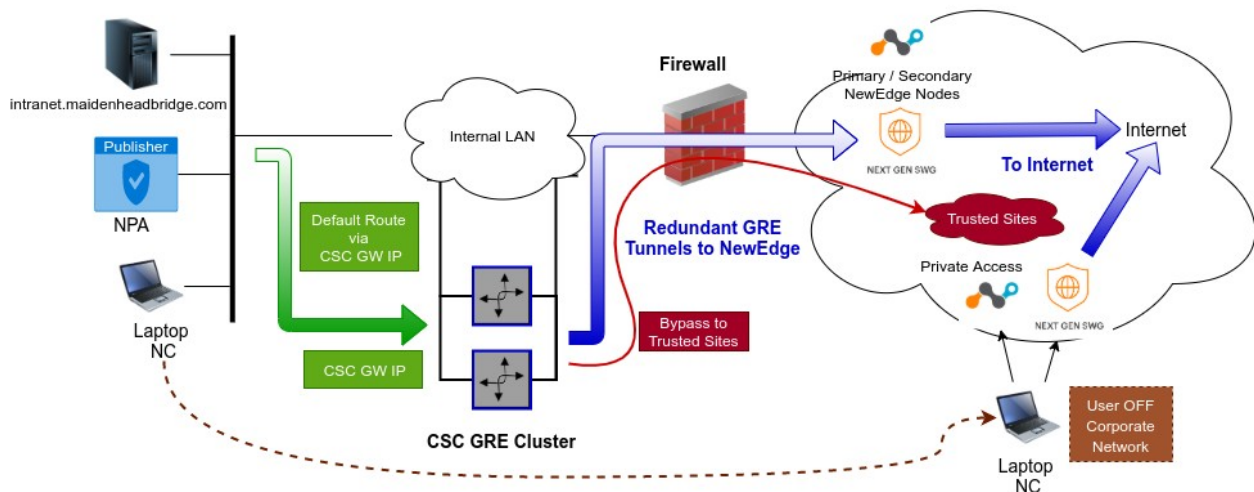
8.1.3.6 Final conclusion

With this setup, the user will have the same experience when connected to the Corporate Office or working from home, thanks to the combination of Maidenhead Bridge Cloud Security Connector and Netskope SWG + Private Access.

8.2 Example of Routed Traffic to Netskope: ON/OFF Corporate Network.

The previous example demonstrated how to create a seamless user experience when using proxy settings on companies devices. In this example, we will reach the same objective but routing the traffic the Netskope and not using proxy settings.

8.2.1 Network diagram



8.2.2 Scenario, Objectives and Solution

Scenario

Before Netskope, the company used a legacy SWG appliance and now wants to migrate to Netskope Cloud SWG + Cloud Firewall. To achieve this purpose, a CSC GRE Cluster will replace the legacy SWG.

The company wants to send all traffic to Netskope. For this reason, the default route to the Internet is via the CSC GRE Cluster.

Objectives

5. To protect Users when surfing the Internet.
6. To provide users with the same experience when they are ON or OFF Corporate Network.
7. To reach Microsoft O365 Authentication destinations using the company Public IP to apply Conditional Access rules and Multifactor Authentication controls.
8. To reach specific sites from the company Public IP.

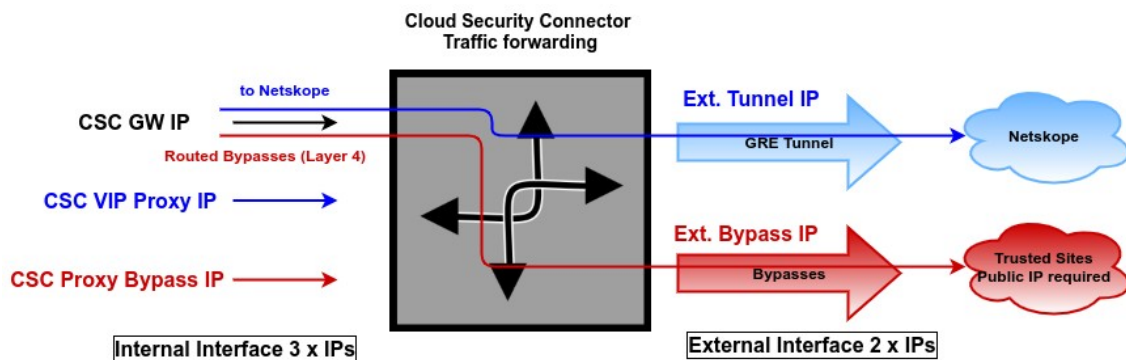
Solution

1. Deploy the CSC GRE Cluster in the same place as the Legacy SWG.
2. Route all traffic via the CSC GRE Cluster.
3. Create the Routed Bypass Rules on the CSC GRE Cluster to reach sites that requires the company Public IP.
4. Use the Netskope Client to steer traffic when the user is OFF the Corporate network and allow remote access to company applications.

8.2.3 Detailed configuration

In the routing scenario, we will use the CSC Gateway IP only.

The flow of the traffic via the CSC will be the following:



CSC internal IP Name:	Proxy IP	Purpose
CSC Gateway IP	172.19.0.60	Traffic to Netskope and Routed Bypasses.

8.2.3.1 Sites to Bypass

Sites required to be bypassed from Netskope in this example:

#	URL	Network / Subnet	Purpose
1	login.microsoftonline.com, login.microsoft.com, login.windows.net	20.190.128.0/18 40.126.0.0/18	O365 login URLs. Bypass required for Conditional Access Rules and MFA controls.
2	portquiz.net	52.47.209.216/32	Trusted site required to be reached via company Public IP.

8.2.3.2 Configuring the Routed Bypass Rules on the CSC

The recommended method to configure Routed Bypass Rules is to create a JSON file and host the JSON file somewhere on the internet or internally on company servers.

In this example, we are hosting the JSON file on an AWS bucket. Here are the details.

JSON URL	https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json
JSON content	<pre>{ "routedBypassRules": [{ "description": "O365 Login URLs 1", "ipProtocol": "tcp", "sourceCirdIp": "0.0.0.0/0", "destinationCirdIp": "20.190.128.0/18", "fromPort": "80", "toPort": "80" }, { "description": "O365 Login URLs 2", "ipProtocol": "tcp", "sourceCirdIp": "0.0.0.0/0", "destinationCirdIp": "20.190.128.0/18", "fromPort": "443", "toPort": "443" }, { "description": "O365 Login URLs 3", "ipProtocol": "tcp", "sourceCirdIp": "0.0.0.0/0", "destinationCirdIp": "40.126.0.0/18", "fromPort": "80", "toPort": "80" }, { "description": "O365 Login URLs 4", "ipProtocol": "tcp", "sourceCirdIp": "0.0.0.0/0", "destinationCirdIp": "40.126.0.0/18", "fromPort": "443", "toPort": "443" }, { "description": "portquiz.net", "ipProtocol": "tcp", "sourceCirdIp": "0.0.0.0/0", "destinationCirdIp": "52.47.209.216/32", "fromPort": "80", "toPort": "80" }] }</pre>

The Routed Bypass functionality allows you to create rules per Source and Destination Network, Protocol (UDP, TCP, ICMP) and Port range.

In this particular case, the rules are:

Rule description	Purpose
O365 Login URLs 1 and O365 Login URLs 2	Routed Bypass to Destination 20.190.128.0/18 port 80 and 433.
O365 Login URLs 3 and O365 Login URLs 4	Routed Bypass to Destination 40.126.0.0/18 port 80 and 433.
portquiz.net	Routed Bypass to Destination 52.47.209.216/32 port 80

8.2.3.3 Netskope Client & Netskope Private Access

The last task is to install the Netskope Client and to configure three things:

1. Steering Configuration. (same than the previous example)
2. Detection ON/OFF Premises. (same than the previous example)
3. Create a Rules for Private Access. In this case, we need to create Rules for all destinations⁽⁹⁾ we want to reach via the company Public IP: O365 Login URLs and "portquiz.net".

New Private App

Private apps are blocked by default. Policies are required to log events and enable access.

APPLICATION NAME

Office 356 Login Networks

+ ADD

HOST

40.126.0.0/18

20.190.128.0/18

PROTOCOL & PORT

TCP: 80,443

UDP: Enter port or port range separated by commas (e.g. 443, 8080-8090)

PUBLISHER

Publishers = MHB-HQ-Publisher

DNS RESOLUTION

Minimum Publisher version of 1.4.6074 is required.

Use Publisher DNS

CANCEL

SAVE AND CREATE POLICY

SAVE

Edit Private App

Private apps are blocked by default. Policies are required to log events and enable access.

APPLICATION NAME

[portquiz.net]

+ ADD

HOST

portquiz.net

PROTOCOL & PORT

TCP: 80

UDP: Enter port or port range separated by commas (e.g. 443, 8080-8090)

PUBLISHER

Publishers = MHB-HQ-Publisher

DNS RESOLUTION

Minimum Publisher version of 1.4.6074 is required.

Use Publisher DNS

CANCEL

SAVE

After the creation of the Private App, you need to include them on Policies. (Go to Policies -> Real Time Protection -> New Policy -> Private App Access.)

Here an example :

- 9 The amount of Private Access Appss to configure is the main difference between the Proxied and Routed solution. In the Proxied solution, it is required to create only one Private Access App: for the CSC Bypass IP. In the Routed solution, it is needed to create a Private Access App for each bypass destination.

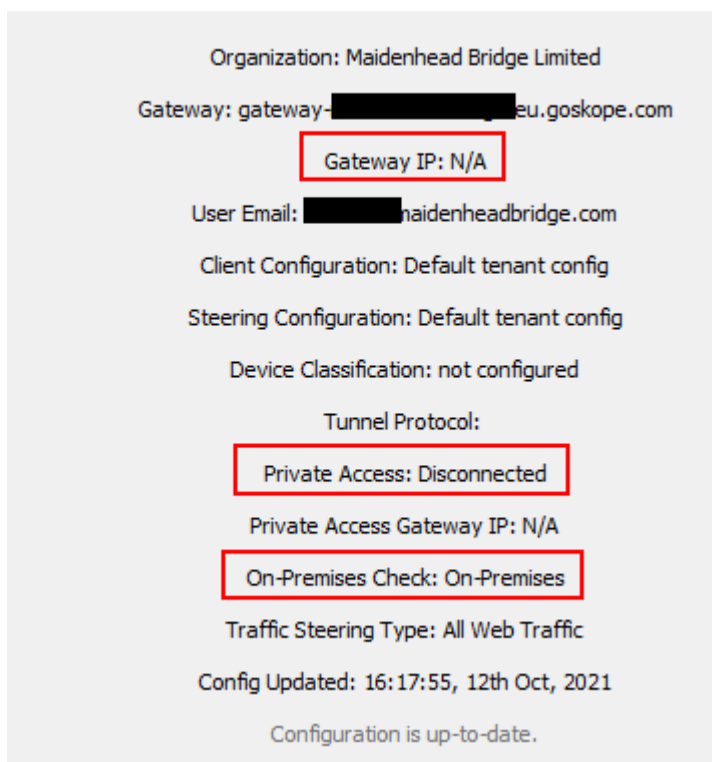


8.2.3.4 User experience On-Prem and Off-Prem

On-Premises

When the User is On-Premises, the Netskope Client icon will show "Netskope Client disabled - GRE detected". Right Click the icon and click Configuration

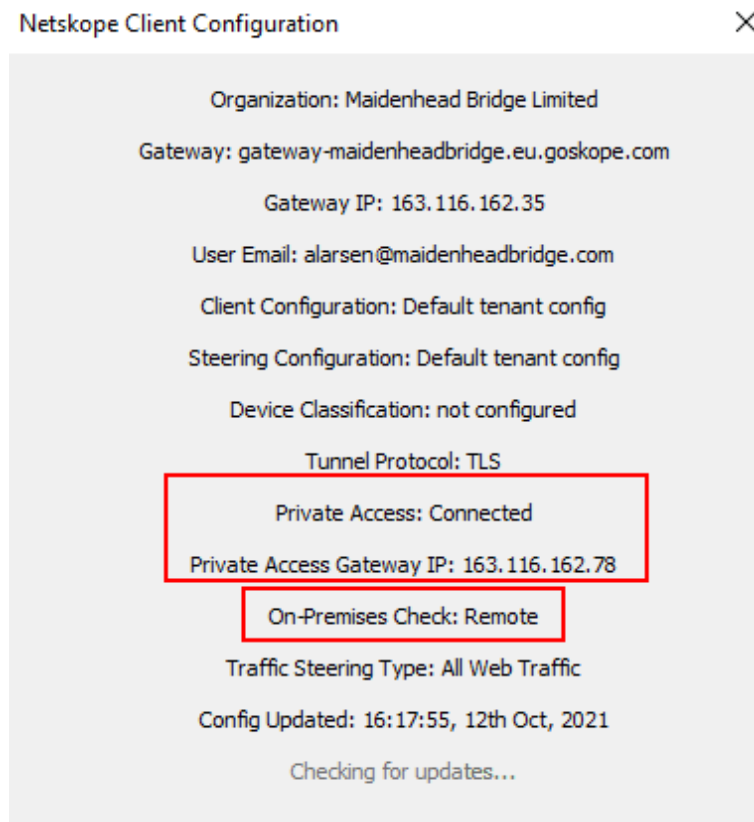
Netskope Client Configuration



When On-Prem, the Netskope client is disabled, the internal LAN network routes all traffic to the Internet to the Cloud Security Connector. The CSC will route the bypass traffic to the Firewall, and everything else will travel via the GRE tunnel to Netskope NewEdge. In this case, Netskope NewEdge provides Cloud Security Web Gateway and Cloud Firewall protection.


Off-Premises

When the User is Off-Premises, the Netskope Client icon will show "Netskope enabled". Right Click the icon and click Configuration:



In this case, the user traffic to the Internet is protected using the Netskope client tunnel, and Private Access is connected, allowing the bypasses to go via the company public IP. How? Below is the explanation.

1. The user types "portquiz.net" on the Browser.
2. The Browser does a DNS query and resolves "portquiz.net" to obtain the destination IP: 52.47.209.216.
3. The Browser initiates the communication to 52.47.209.216
4. Private Access intercepts the communication to "Private App" -> 52.47.209.216 (port 80) and sends the traffic to the Publisher that is on the company behind the CSC.

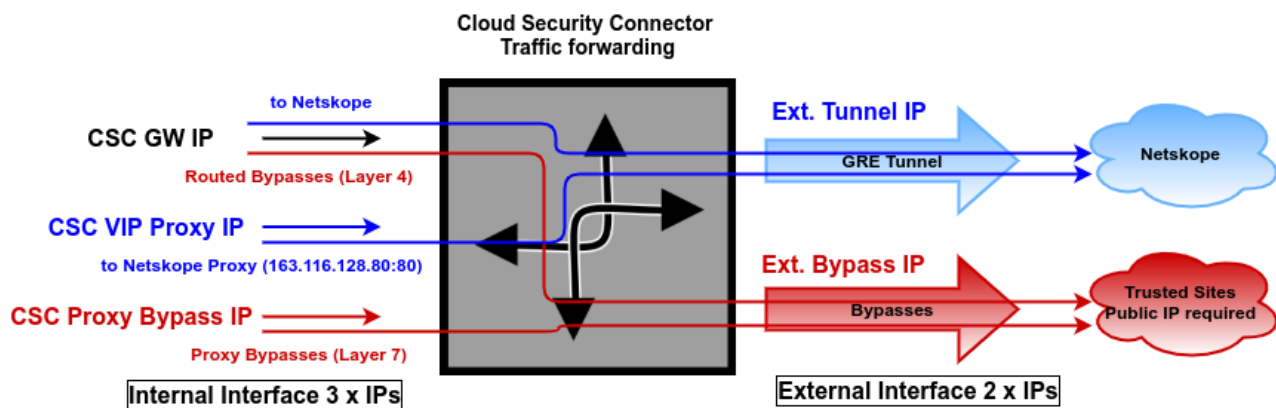
- 
5. The "default route" sends the flow to the CSC. The CSC checks the destination IP and Port (52.47.209.216, port 80) and finds it on the "Routed Bypass List". Finally, the flow leaves to the Internet via the Corporate Public IP.

8.2.3.5 *Final conclusion*

With this setup, the user will have the same experience when connected to the Corporate Office or working from home, thanks to the combination of Maidenhead Bridge Cloud Security Connector and Netskope.

8.3 Putting all together: Proxied and Routed Environments.

The Cloud Security Connector covers all possible scenarios, and this is the main benefit. The Cloud Security Connector will protect all company devices with maximum redundancy and High Availability. Below is a list of real cases of devices to connect in large organizations.



This table shows some examples of devices and the steering traffic options for each one.

#	Device	Steering traffic options ⁽¹⁰⁾
1	Laptops	Netskope Client (NC), PAC files , Routing.
2	Desktops	Netskope Client (NC), PAC files , Routing, Explicit Proxy.
3	Servers	PAC Files, Routing, Explicit Proxy.
4	Servers with no default route to Internet.	PAC files, Explicit Proxy.
5	Companies with No Default Route to Internet. All devices.	PAC files, Explicit Proxy.
6	IOT Devices.	Routing, Explicit Proxy.
7	Linux Servers.	Routing, Explicit Proxy.
8	Cameras, Room Reservations Systems, Alarms Systems, etc.	Routing, Explicit Proxy.

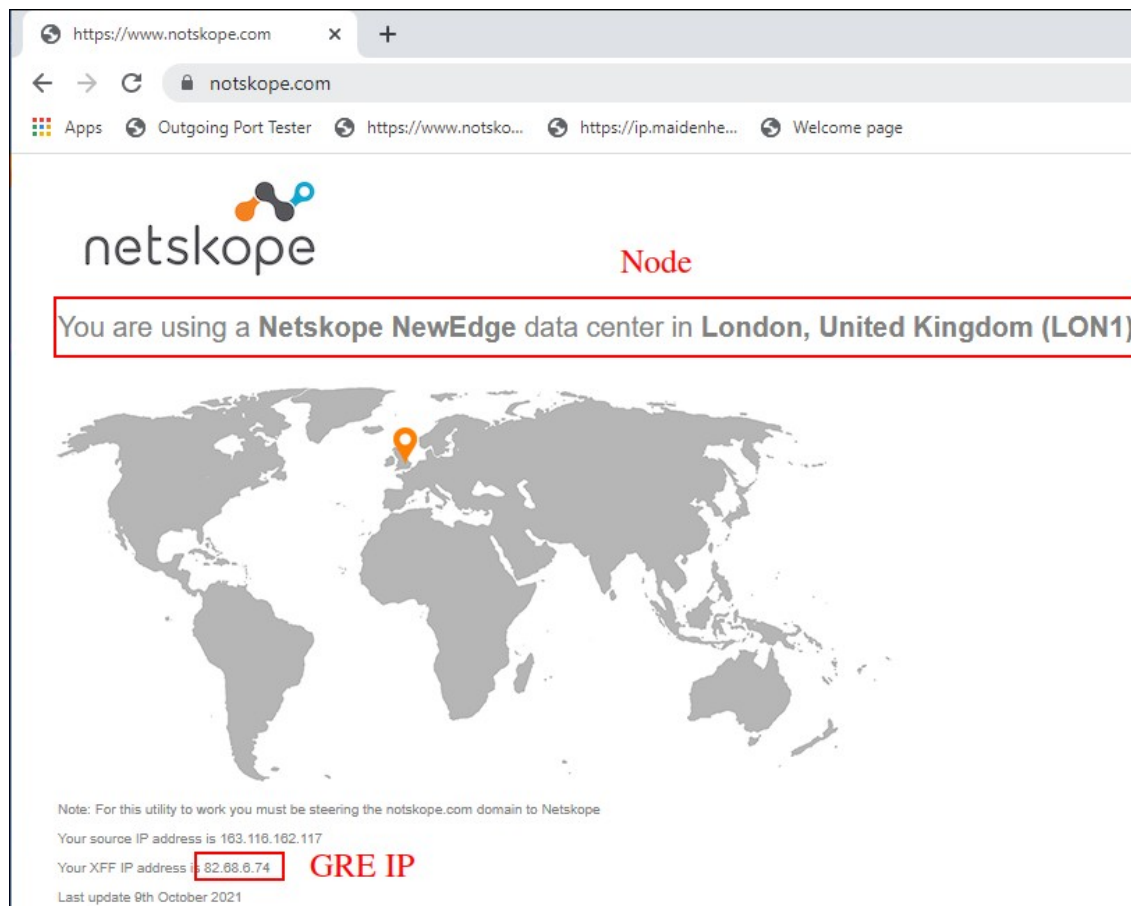
¹⁰ In most cases, you can combine the options shown.

8.4 Testing traffic to Netskope

8.4.1 www.netskope.com

The page www.netskope.com shows the Netskope Datacenter and the GRE IP of the CSC.

Using the browser:



Using curl command from CMD or Terminal

Proxy environment:

Command	<code>curl --proxy http://<CSC VIP>:80 www.netskope.com</code> (i.e. <code>\$curl --proxy http://172.19.0.61:80 www.netskope.com</code>)
Expected Result	<NewEdge Node IP> <City>,<Country> (<NodeID>) (i.e. 163.116.162.117 London, United Kingdom (LON1))

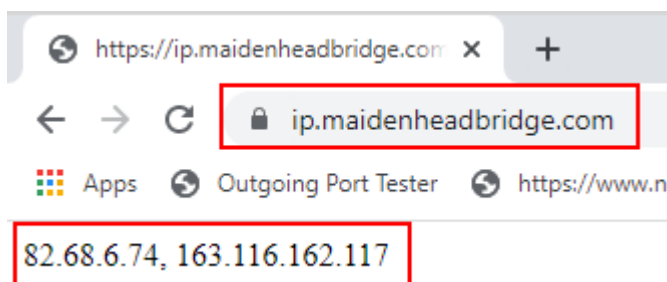
Routed environment:

Command	curl www.netskope.com (i.e. \$curl www.netskope.com)
Expected Result	<NewEdge Node IP> <City>, <Country> (<NodeID>) (i.e. 163.116.162.117 London, United Kingdom (LON1))

8.4.2 https://ip.maidenheadbridge.com

Maidenhead Bridge provides a HTTPS page to check the IP.

Using the Browser:



Using curl command from CMD or Terminal

(Note: the switch "-k" on curl command is to avoid SSL certificate validation)

Proxy environment:

Command	curl -k --proxy http://<CSC VIP>:80 https://ip.maidenheadbridge.com (i.e. \$curl --proxy http://172.19.0.61:80 https://ip.maidenheadbridge.com)
Expected Result	<Customer IP>, <Netskope Node IP> (i.e. 82.68.6.74, 163.116.162.117)

Routed environment:

Command	curl -k https://ip.maidenheadbridge.com (i.e. \$curl -k https://ip.maidenheadbridge.com)
Expected Result	<Customer IP>, <Netskope Node IP> (i.e. 82.68.6.74, 163.116.162.117)

8.4.3 SpeedTest

The CSC contains the SpeedTest client. You can run it from the SSH console or using any Management tool (AWS Systems Manager, Rundeck, Salt, Ansible, etc.)

```
Monitoring Tasks
1) Show Configuration and Status
2) Show Interfaces Traffic
3) Traceroute and Latency Test
4) Speed Test (Experimental)
```

```
Selection: 4

SPEED TEST
This is experimental. We are using third party tools. (Speedtest.net)
Results can be inaccurate or none. The test takes a while

Retrieving speedtest.net configuration...
Testing from Netskope (163.116.162.116)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by Livedrive Internet (London) [1.12 km]: 26.278 ms
Testing download speed.....
Download: 343.44 Mbit/s
Testing upload speed.....
Upload: 69.89 Mbit/s
```

9 SSH Admin Console

The SSH Admin Console simplifies admin tasks showing what is essential to administrators for configuration, operation and troubleshooting.

Access to SSH Admin Console: `$ssh cscadmin@<CSC GW IP>`

User: **cscadmin** / Default Password: **maidenheadbridge** / IP to SSH <CSC GW IP>

Main Menu:

```
Maidenhead Bridge
Cloud Security Connector GRE cluster for Netskope - Admin Console
Company : Maidenhead Bridge
Location : mhbDcIp74
CSC ID : ns-cgc00004-a
Soft Version : 1.1

Please select an option by typing its number

Monitoring Tasks
1) Show Configuration and Status
2) Show Interfaces Traffic
3) Traceroute and Latency Test
4) Speed Test (Experimental)

CSC Admin tasks
5) AWS SSM Agent (Register or De-Register)
6) Manage Administrators
7) Change Timezone

Proxy Bypass
8) View Current Proxy Bypass List
9) Configure Proxy Bypass List

Routed Bypass
10) View Current Routed Bypass List
11) Configure Routed Bypass List

Log Information
12) View Current Month
13) View Last 6 Months

Configuration Wizards
14) Change Nodes, DNS servers, Syslog and more.
15) Switch Tunnels - Primary / Secondary.
16) Update Netskope Nodes Databases.

MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.

e) Exit

Selection: 
```


9.1 Monitoring Tasks

Monitoring Tasks

- 1) Show Configuration and Status
- 2) Show Interfaces Traffic
- 3) Traceroute and Latency Test
- 4) Speed Test (Experimental)

9.1.1 Show Configuration and Status

Show Configuration and Status. This menu show all parameters configured on the CSC GRE and does several checks.

```
GENERAL INFORMATION
Company : Maidenhead Bridge
Location : HQkvm
CSC ID : ns-cgc00002-a
CSC date: Wed 13 Oct 14:12:20 BST 2021
Soft version : 1.0

INTERFACES INFORMATION
External: Tunnel IP: 192.168.1.60 | Bypass Proxy Egress IP: 192.168.1.61 | CSC IP(eth0): 192.168.1.62/24 | Network Gateway: 192.168.1.240 is Alive
Internal: CSC GW IP: 172.19.0.60 | CSC IP(eth1): 172.19.0.63/24 | Network Gateway: 172.19.0.133 is Alive

TRAFFIC REDIRECTION Options
To Netskope: VIP Proxy: 172.19.0.61:80 | Route all traffic via CSC GW IP | Netskope Global Proxy IP: 163.116.128.80:80 via CSC GW IP
Direct to Internet: Bypass Proxy: 172.19.0.62:3128 | Netskope Global Proxy IP: 163.116.128.80:3128 via CSC GW IP

DNS INFORMATION
DNS Server (1) IP: 172.19.0.100 is Alive
DNS Server (2) IP: 1.1.1.1 is Alive

NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74

Primary Tunnel:
  Node : GB,London,LON1
  Node Public IP: 163.116.162.36
  Node Probe: 10.162.6.209

Secondary Tunnel:
  Node : GB,Manchester,MAN1
  Node Public IP: 163.116.165.36
  Node Probe: 10.165.6.209

TUNNEL STATUS
Primary Tunnel (reachability):
  Node Keepalive is: Alive
  GRE Tunnel IP is: Alive

Secondary Tunnel (reachability):
  Node Keepalive is: Alive
  GRE Tunnel IP is: Alive

returnToPrimaryTunnel: true

Tunnel Status: Primary tunnel is active since: Fri 8 Oct 03:19:08 BST 2021

HTTP://WWW.NOTSKOPE.COM PAGE STATUS
163.116.162.116 London, United Kingdom (LON1)

PROXY BYPASS - EGRESS INTERFACE STATUS
Proxy Bypass Egress Interface 192.168.1.61 can reach test page (https://ip.maidenheadbridge.com) via Public IP 82.68.6.73

ROUTED BYPASS
Using Routed Bypass URL: https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json
Routed Bypass URL https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json is reachable
Routed Bypass Rules configured via URL: 8

AWS SSM AGENT
AWS SSM Agent is active (running) since Tue 2021-10-05 20:23:47 BST; 1 weeks 0 days ago
Registration values: {"ManagedInstanceID":"mi-0160555d766bf22c6","Region":"eu-west-2"}

SYSLOG INFORMATION
SYSLOG Server (1) IP: 172.19.0.199 is Alive
SYSLOG Server (2) IP is not configured
SYSLOG TCP Port: 514

HIGH AVAILABILITY Information
This CSC (ns-cgc00002-a) is Cluster ACTIVE
```

9.1.1.1 General Information

```
GENERAL INFORMATION
Company : Maidenhead Bridge
Location : HQkvm
CSC ID : ns-cgc00002-a
CSC date: Wed 13 Oct 14:27:33 BST 2021
Soft version : 1.0
```

9.1.1.2 Interfaces Information

```
INTERFACES INFORMATION
External: Tunnel IP: 192.168.1.60 | Bypass Proxy Egress IP: 192.168.1.61 | CSC IP(eth0): 192.168.1.62/24 | Network Gateway: 192.168.1.240 is Alive
Internal: CSC GW IP: 172.19.0.60 | CSC IP(eth1): 172.19.0.63/24 | Network Gateway: 172.19.0.133 is Alive
```

9.1.1.3 TRAFFIC REDIRECTION Options

```
TRAFFIC REDIRECTION Options
To Netskope: VIP Proxy: 172.19.0.61:80 | Route all traffic via CSC GW IP | Netskope Global Proxy IP: 163.116.128.80:80 via CSC GW IP
Direct to Internet: Bypass Proxy: 172.19.0.62:3128 | Netskope Global Proxy IP: 163.116.128.80:3128 via CSC GW IP
```

9.1.1.4 DNS INFORMATION

```
DNS INFORMATION
DNS Server (1) IP: 172.19.0.100 is Alive
DNS Server (2) IP: 1.1.1.1 is Alive
```

9.1.1.5 NETSKOPE INFORMATION

```
NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74

Primary Tunnel:
    Node : GB,London,LON1
    Node Public IP: 163.116.162.36
    Node Probe: 10.162.6.209
Secondary Tunnel:
    Node : GB,Manchester,MAN1
    Node Public IP: 163.116.165.36
    Node Probe: 10.165.6.209
```

9.1.1.6 TUNNEL STATUS

```
TUNNEL STATUS
Primary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
Secondary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
returnToPrimaryTunnel: true
Tunnel Status: Primary tunnel is active since: Fri 8 Oct 03:19:08 BST 2021
```

9.1.1.7 HTTP://WWW.NOTSKOPE.COM PAGE STATUS

```
HTTP://WWW.NOTSKOPE.COM PAGE STATUS
163.116.162.116 London, United Kingdom (LON1)
```

9.1.1.8 PROXY BYPASS - EGRESS INTERFACE STATUS

```
PROXY BYPASS - EGRESS INTERFACE STATUS
Proxy Bypass Egress Interface 192.168.1.61 can reach test page (https://ip.maidenheadbridge.com) via Public IP 82.68.6.73
```

9.1.1.9 ROUTED BYPASS

```
ROUTED BYPASS
Using Routed Bypass URL: https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json
Routed Bypass URL https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json is reachable
Routed Bypass Rules configured via URL: 8
```

9.1.1.10 AWS SSM AGENT

```
AWS SSM AGENT
AWS SSM Agent is active (running) since Tue 2021-10-05 20:23:47 BST; 1 weeks 0 days ago
Registration values: {"ManagedInstanceID":"mi-0160555d766bf22c6","Region":"eu-west-2"}
```

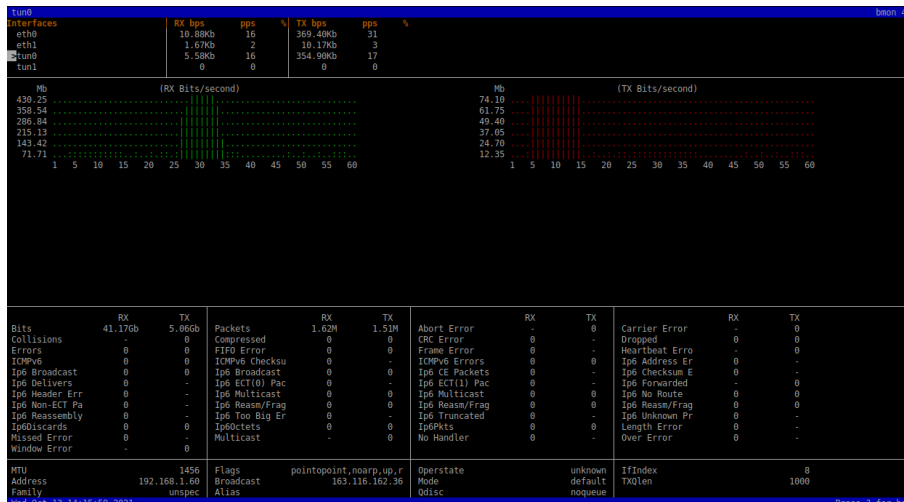
9.1.1.11 SYSLOG INFORMATION

```
SYSLOG INFORMATION
SYSLOG Server (1) IP: 172.19.0.199 is Alive
SYSLOG Server (2) IP is not configured
SYSLOG TCP Port: 514
```


9.1.1.12 HIGH AVAILABILITY Information

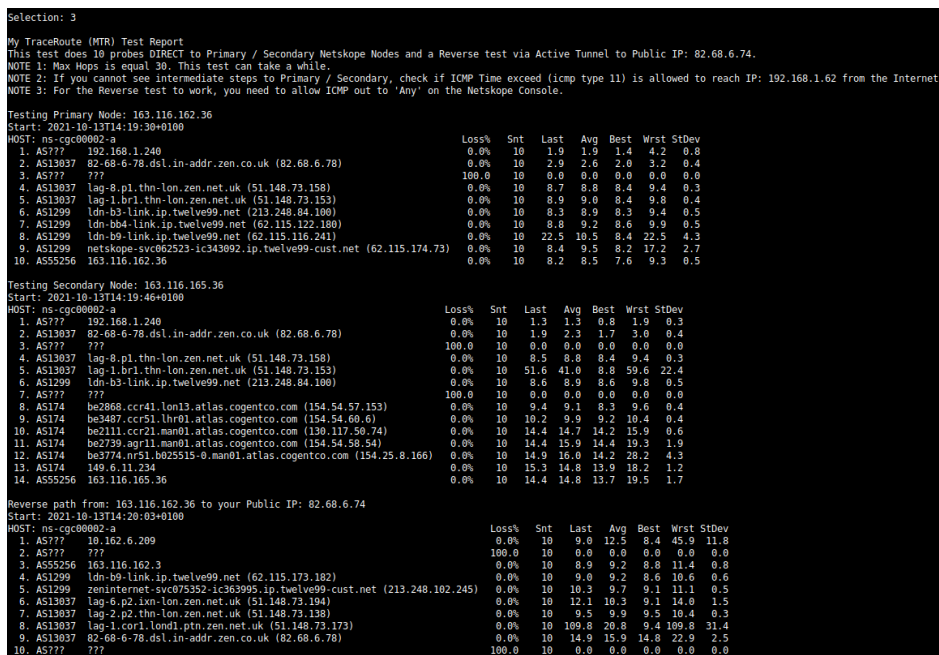
HIGH AVAILABILITY Information
This CSC (ns-cgc00002-a) is Cluster ACTIVE

9.1.2 Show Interfaces Traffic



9.1.3 Traceroute and Latency Test

This test does a MyTraceRoute test to Primary and Secondary Node and a reverse traceroute from the active tunnel to the local IP. This test helps administrators to validate the quality of the Internet path, hop by hop.



9.1.4 Speed Test

```
Selection: 4

SPEED TEST
This is experimental. We are using third party tools. (Speedtest.net)
Results can be inaccurate or none. The test takes a while

Retrieving speedtest.net configuration...
Testing from Netskope (163.116.162.116)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by Lightning Fibre Ltd (London) [1.12 km]: 27.268 ms
Testing download speed.....
Download: 369.82 Mbit/s
Testing upload speed.....
Upload: 70.33 Mbit/s
```

9.2 CSC Admin Tasks

```
CSC Admin tasks
5) AWS SSM Agent (Register or De-Register)
6) Manage Administrators
7) Change Timezone
```

9.2.1 AWS SSM Agent (Register / De-Register)

The CSC GRE can be integrated as "Managed Instance" with Amazon Cloud (AWS).

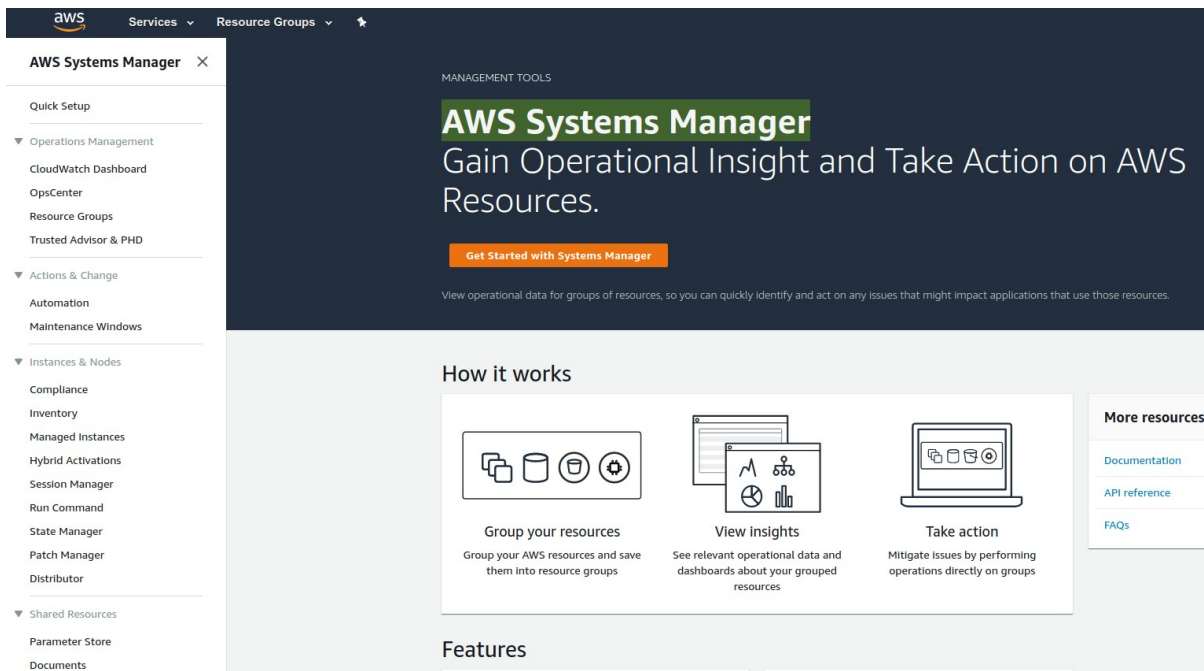
Amazon AWS offers on the Free Tier Account (<https://aws.amazon.com/free>) the capability to add up to 1000 managed instances.

The steps required to add the CSC to AWS are two:

1. Create the Keys to register using "Hybrid Activation".
2. Register the CSC with the Keys

9.2.1.1 *Create the Key using "Hybrid Activations"*

1. Open your AWS console and go to: "AWS Systems Manager"



- Click "Hybrid Activations". We recommend to put the name to identify the CSC on "Activation Description" and "Default Instance Name". In this example is cgs00045

3. Click "Create activation" to generate the Keys. Please, also note the AWS Region

eu-west-2.console.aws.amazon.com/systems-manager/activations/?region=eu-west-2 3

AWS Systems Manager

Quick Setup

Operations Management

CloudWatch Dashboard

OpsCenter

Resource Groups

Trusted Advisor & PHD

Actions & Change

Automation

Maintenance Windows

Instances & Nodes

You have successfully created a new activation. Your activation code is listed below. Copy this code and keep it in a safe place as you will need it to install the Amazon SSM Agent on your instances. You can now install amazon-ssm-agent and manage your instance using Run Command. [Learn more](#)

Activation Code HL7upb+rwnrMd+cln+4p 1

Activation ID d3f3ddf7-23fb-4b3e-9778-2af6e09a1f95 2

AWS Systems Manager > Activations

Activations

ID	Description	Registered instances	Registered instances count
d3f3ddf7-23fb-4b3e-9778-2af6e09a1f95	cgs00045	0	1

9.2.1.2 Register the CSC on AWS

Using the Keys and Region from the Step before, register the CSC.

1. From the CSC Admin Tasks Menu, select "5) AWS SSM Agent (Register or De-Register)"

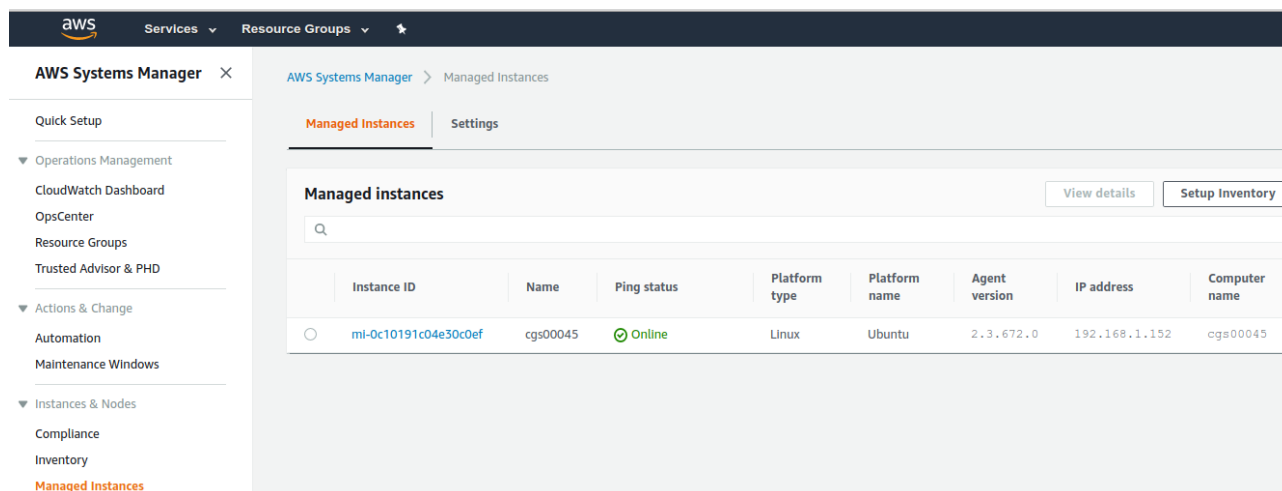
```
CSC Admin tasks
5) AWS SSM Agent (Register or De-Register)
6) Manage Administrators
7) Change Timezone
```

2. Register using the Keys and region:

```
Selection: 5 1
AWS SSM Agent is not registered
Do you want to Register (start) the AWS SSM Agent (y/n) y 2
Please, ingress Activation Code, Activation ID and Region (example: eu-west-1)
Activation Code :HL7upb+rwnrMd+cln+4p 3
Activation ID :d3f3ddf7-23fb-4b3e-9778-2af6e09a1f95 4
Region :eu-west-2 5
AWS SSM Agent is active (running) since Sat 2019-08-24 14:35:57 UTC; 22ms ago
Registration values: {"ManagedInstanceID":"mi-0c10191c04e30c0ef","Region":"eu-west-2"} 6
```

Done! You have the CSC integrated with AWS now with the instance-id "mi-xxxxxxx" ("mi-0c10191c04e30c0ef" in this example).

Go to AWS System Manager → Managed Instances you will be able to see your CSC.



9.2.1.3 Checking the status of the AWS SSM agent

The "Show Configuration and Status" Menu shows the status of the AWS SSM agent at the bottom.

```
AWS SSM AGENT
AWS SSM Agent is active (running) since Sat 2019-08-24 14:35:57 UTC; 7min ago
Registration values: {"ManagedInstanceID":"mi-0c10191c04e30c0ef","Region":"eu-west-2"}
```

IMPORTANT: Go to Appendix B to learn how to "Run Commands" from the AWS console to monitoring the CSC and Update Bypass Lists.

9.2.2 Manage Administrators

This section allows to setup specific settings for user "cscadmin" (console user) and "csccli" (cli user).

```
Selection: 6
Please, select the Administrator: 'cscadmin' or 'csccli'
1) cscadmin
2) csccli
3) Quit
Enter your choice: █
```

9.2.2.1 cscadmin settings

```
Enter your choice: 1
Please, select the task to do:
1) Change Password
2) Manage SSH Keys
3) Quit
Enter your choice: █
```

9.2.2.2 csccli user

Note: the "csccli" user allows console access to the CSC. If you are managing the CSC using Rundeck, Salt or Ansible, you will need to enable the "csccli" user and to setup the SSH Key.

```
Enter your choice: 2
User 'csccli' is enabled.
Please, select the action to take.
1) Disable csccli User
2) Change SSH Key
3) Quit
Enter your choice: █
```

9.2.3 Change Timezone

You can change the TimeZone of the CSC using this menu.

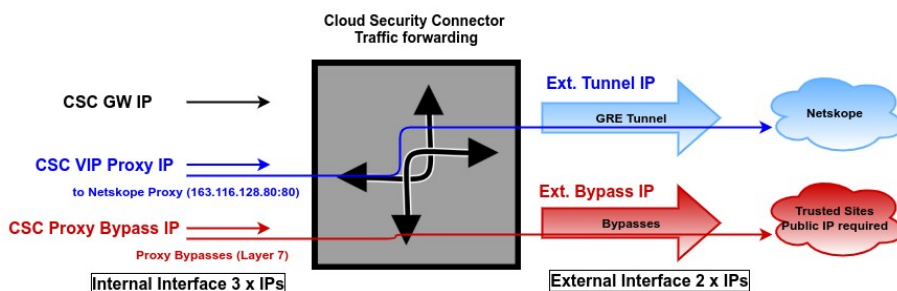


9.3 Proxy Bypass

The Proxy Bypass functionality allows doing layer 7 bypasses. This functionality works in conjunction with PAC files.

```
Proxy Bypass
8) View Current Proxy Bypass List
9) Configure Proxy Bypass List
```

9.3.1 Proxy Bypass - Traffic Flow



9.3.2 View Current Proxy Bypass List

This menu displays the current Proxy Bypass List. For example:

```
Selection: 8
This is the list of current Domains configured:
login.microsoftonline.com
login.microsoft.com
login.windows.net
portquiz.net
Press Enter to continue...
```

9.3.3 Configure Proxy Bypass List

This menu allows to configure the Proxy Bypass List.

```
Please, select method to configure Proxy Bypass List
1) Auto - Proxy Bypass PAC URL
2) Manual
3) Quit
Enter your choice: █
```

9.3.3.1 Auto - Proxy Bypass PAC URL

Auto-Proxy Bypass PAC URL is the recommended method to use. You need to create a "Proxy Bypass PAC file" and host the PAC file somewhere on the internet. The CSC will read the "Proxy Bypass List" from the "Proxy Bypass PAC file".

The "Proxy Bypass PAC file" acts as a central repository of all Layer 7 bypasses required. Moreover, if you manage the CSCs using AWS Systems Manager (or another tool), you can update all CSCs in your network doing one command.

In Chapter 8.1) "Example of Proxied Traffic to Netskope: ON/OFF Corporate Network. " we explained how to use the Proxy Bypass functionality. In this chapter, we are going to see how to configure and refresh the list.

Example of Proxy Bypass PAC:

```
Please, select method to configure Proxy Bypass List
1) Auto - Proxy Bypass PAC URL
2) Manual
3) Quit
Enter your choice: 1

Please, select method to configure Proxy Bypass List
1) Configure Proxy Bypass PAC URL and/or Update Proxy Bypasses
2) See PAC Proxy Bypass Example
3) Quit
Enter your choice: 2

function FindProxyForURL(url, host) {
    var bypassproxy="PROXY 1.1.1.1:3128; PROXY 2.2.2.2:3128";

    // =====
    // Section 3: Bypass via Cloud Security Connectors

    // Bypass via CSC Public IPs (Examples)
    // Okta Domains (for Location Rules)
    if ((shExpMatch(host, "*.okta.com")) ||
        (shExpMatch(host, "*.oktacdn.com")) ||
        (shExpMatch(host, "*.okta-emea.com")) ||
        (shExpMatch(host, "login.mydomain.com"))) ||
        // 0365 Domains for ConditionalAccess
        (shExpMatch(host, "login.microsoftonline.com"))) ||
        (shExpMatch(host, "login.microsoft.com"))) ||
        (shExpMatch(host, "login.windows.net"))) ||
        // IP Test Page
        (shExpMatch(host, "ip.maidenheadbridge.com")))) {
        return bypassproxy
    }

    return bypassproxy
}
```

Note 1: It is mandatory to use this function and format. Feel free to add lines but don't change the format. We recommend to start filling the first line and the last line. Use middle lines for copy/paste.

Note 2: The Bypass Proxy port is 3128

Configuring the Proxy Bypass PAC URL and Refresh the List

```
Selection: 9
Please, select method to configure Proxy Bypass List
1) Auto - Proxy Bypass PAC URL
2) Manual
3) Quit
Enter your choice: 1
Please, select method to configure Proxy Bypass List
1) Configure Proxy Bypass PAC URL and/or Update Proxy Bypasses
2) See PAC Proxy Bypass Example
3) Quit
Enter your choice: 1
Proxy Bypass Configuration
Your current Proxy Bypass PAC URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
Do you want to change the Proxy Bypass PAC URL?
1) Yes
2) No
Enter your choice: 1
Please, input Proxy Bypass PAC URL
Bypass PAC URL:https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
Your current Proxy Bypass PAC URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
Do you want to refresh Proxy Bypass List?
1) Yes
2) No
Enter your choice: 1
This is your current Proxy Bypass List
login.microsoftonline.com
login.microsoft.com
login.windows.net
portquiz.net
Do you want apply changes?
1) Yes
2) No
Enter your choice: 1
Proxy Bypass List updated successfully.
```

9.3.3.2 Manual

If you want to update manually your Proxy Bypass list, follow this steps.

1. Select Option 2)

```
1) Auto - Proxy Bypass PAC URL
2) Manual
3) Quit
Enter your choice: 2
Please, read the instructions carefully:
You are going to edit the list using NANO editor
The following formats are accepted:
Full Domains: 'www.example.com'
Wildcard for subdomains: '.example.com' - This will allow all subdomains of example.com
To save, press CTRL-X and 'Yes'
Paid attention to ERROR messages if any. ERRORS must be corrected before to continue
Do you want to continue? (y/n)?
```

2. Input "y"


```
GNU nano 4.8 domains Modified
.okta.com
.oktacdn.com
.okta-emea.com
login.mydomain.com
login.microsoftonline.com
login.microsoft.com
login.windows.net
portquiz.net
manualAdded.com
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste Text ^I To Spell ^_ Go To Line M-E Redo
```

3. Add / Delete / Modify your full domains and subdomains
4. Please, CTL+X and "Yes" (and after next prompt Enter) to Save
5. The modified Bypass List will be displayed.

```
This is your current Proxy Bypass List

.okta.com
.oktacdn.com
.okta-emea.com
login.mydomain.com
login.microsoftonline.com
login.microsoft.com
login.windows.net
portquiz.net
manualAdded.com

Do you want apply changes?
1) Yes
2) No
Enter your choice: 
```

6. Apply Changes Yes or No. If "1" you will receive the following message:

```
Do you want apply changes?
1) Yes
2) No
Enter your choice: 1

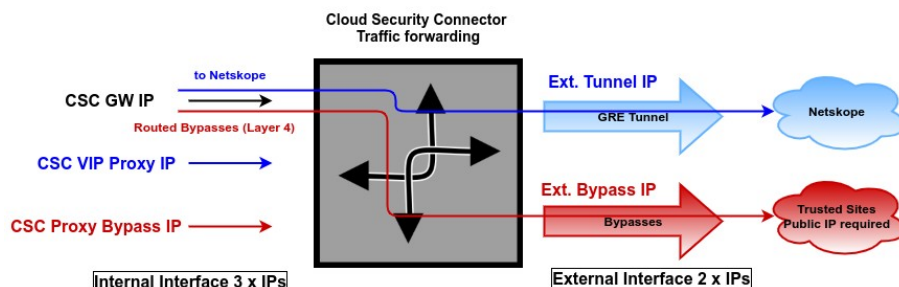
Proxy Bypass List updated sucessfully.
```

9.4 Routed Bypass

When routing all traffic via the CSC GW IP, the Routed Bypass functionality allows you to connect specific destinations (IP/Subnet) direct to the Internet, using your Public IP. By default, all destinations will travel via the GRE tunnel to Netskope. If you want to bypass the GRE tunnel, you need to create a Routed Bypass Rule.

```
Routed Bypass
10) View Current Routed Bypass List
11) Configure Routed Bypass List
```

9.4.1 Routed Bypass - Traffic Flow



9.4.2 View Current Routed Bypass List

You can select to view the Routed Bypass Rules in Compact format or JSON.

```
Selection: 10
Please, Select 'Compact' or 'Json' format
1) Compact
2) Json
3) Quit
Enter your choice: ☐
```

9.4.2.1 Compact

Current Values configured are:

```
Index: 0, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 1"
Index: 1, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 2"
Index: 2, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 3"
Index: 3, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 52.47.209.216/32, FromPort: 80, To Port: 80, Description: "portquiz.net"
Index: 4, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 4"
Index: 5, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 13.107.64.0/18, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 1"
Index: 6, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.112.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 2"
Index: 7, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.120.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 3"
```

9.4.2.2 Json

```
Enter your choice: 2
{
  "routedBypassRules": [
    {
      "description": "0365 Login URLs 1",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "20.190.128.0/18",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "0365 Login URLs 2",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "20.190.128.0/18",
      "fromPort": "443",
      "toPort": "443"
    },
    {
      "description": "0365 Login URLs 3",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "40.126.0.0/18",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "portquiz.net",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.47.209.216/32",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "0365 Login URLs 4",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "40.126.0.0/18",
      "fromPort": "443",
      "toPort": "443"
    },
    {
      "description": "Skype and Teams UDP 1",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "13.107.64.0/18",
      "fromPort": "3478",
      "toPort": "3481"
    },
    {
      "description": "Skype and Teams UDP 2",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.112.0.0/14",
      "fromPort": "3478",
      "toPort": "3481"
    },
    {
      "description": "Skype and Teams UDP 3",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.120.0.0/14",
      "fromPort": "3478",
      "toPort": "3481"
    }
  ]
}
Press ENTER to continue
```


9.4.3 Configure Routed Bypass List

There are two methods to configure the Routed Bypass List: Routed Bypass URL and Manual. The recommended method is to use Routed Bypass URL.

```
Selection: 11

Please, Select Method:
1) Routed Bypass URL
2) Manual (Paste Routed Bypass Rules JSON File)
3) Reset to Default Values
4) Quit
Enter your choice: █
```

9.4.3.1 Routed Bypass URL

Routed Bypass URL is the recommended method. Create an AWS bucket and place your JSON file on it. Here an example:

<https://mhbm-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json>

```
Enter your choice: 1
Your Routed Bypass URL configured is: https://csc-gre-aws.s3.amazonaws.com/routedBypassRulesFile.json
Do you want to change the Routed Bypass URL?
1) Yes
2) No
Enter your choice: 1
Please, input Routed Bypass URL
Routed Bypass URL: https://csc-gre-aws.s3.amazonaws.com/routedBypassRulesFile.json
Do you want to refresh the Routed Bypass List?
1) Yes
2) No
Enter your choice: 1
Routed Bypass JSON file imported successfully

You can review your values before to apply. Please, Select 'Compact' or 'Json' format.
1) Compact
2) Json
3) No review is needed
Enter your choice: 1
Current Values configured are:
Index: 0, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 1"
Index: 1, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 2"
Index: 2, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 3"
Index: 3, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 52.47.209.216/32, FromPort: 80, To Port: 80, Description: "portquiz.net"
Index: 4, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 4"
Index: 5, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 13.107.64.0/18, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 1"
Index: 6, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.112.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 2"
Index: 7, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.120.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 3"
Do you want to apply this values?
1) Yes
2) No
Enter your choice: 1
(Index: 0) Rule "0365 Login URLs 1" was created successfully.
(Index: 1) Rule "0365 Login URLs 2" was created successfully.
(Index: 2) Rule "0365 Login URLs 3" was created successfully.
(Index: 3) Rule "portquiz.net" was created successfully.
(Index: 4) Rule "0365 Login URLs 4" was created successfully.
(Index: 5) Rule "Skype and Teams UDP 1" was created successfully.
(Index: 6) Rule "Skype and Teams UDP 2" was created successfully.
(Index: 7) Rule "Skype and Teams UDP 3" was created successfully.
Routed Bypass List updated successfully.
Press ENTER to continue
```

9.4.3.2 Manual (Paste Routed Bypass JSON file)

Another option to configure Routed Bypass Rules is to paste the JSON file using the following menu:

```
Enter your choice: 2

WARNING: Manual Configuration will remove the Bypass Routed URL if configured.

Do you want to paste the Routed Bypass Rules JSON File?
1) Yes
2) No
Enter your choice: 1

Please, paste Routed Bypass JSON File and press 'Enter' if required.

NOTE: If the json file has errors, it is possible that the script will hang. Press '}' and 'Enter' to end the operation.

Routed Bypass JSON file: }
```

and paste the JSON file. The JSON file will be displayed, and if no errors are found, you can apply the changes:

```
{
  "description": "Skype and Teams UDP 3",
  "ipProtocol": "udp",
  "sourceCirdIp": "0.0.0.0/0",
  "destinationCirdIp": "52.120.0.0/14",
  "fromPort": "3478",
  "toPort": "3481"
}

Routed Bypass JSON file imported successfully

You can review your values before to apply. Please, Select 'Compact' or 'Json' format.
1) Compact
2) Json
3) No review is needed
Enter your choice: 1

Current Values configured are:

Index: 0, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 1"
Index: 1, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 20.190.128.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 2"
Index: 2, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 80, To Port: 80, Description: "0365 Login URLs 3"
Index: 3, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 52.47.209.216/32, FromPort: 80, To Port: 80, Description: "portquiz.net"
Index: 4, Protocol: tcp, SourceIP: 0.0.0.0/0, DestinationIP: 40.126.0.0/18, FromPort: 443, To Port: 443, Description: "0365 Login URLs 4"
Index: 5, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 13.107.64.0/18, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 1"
Index: 6, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.112.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 2"
Index: 7, Protocol: udp, SourceIP: 0.0.0.0/0, DestinationIP: 52.120.0.0/14, FromPort: 3478, To Port: 3481, Description: "Skype and Teams UDP 3"

Do you want to apply this values?
1) Yes
2) No
Enter your choice: 1

(Index: 0) Rule "0365 Login URLs 1" was created succesfully.
(Index: 1) Rule "0365 Login URLs 2" was created succesfully.
(Index: 2) Rule "0365 Login URLs 3" was created succesfully.
(Index: 3) Rule "portquiz.net" was created succesfully.
(Index: 4) Rule "0365 Login URLs 4" was created succesfully.
(Index: 5) Rule "Skype and Teams UDP 1" was created succesfully.
(Index: 6) Rule "Skype and Teams UDP 2" was created succesfully.
(Index: 7) Rule "Skype and Teams UDP 3" was created succesfully.

Routed Bypass List updated succesfully.

Press ENTER to continue
```

9.5 Log Information

This section shows the Logs. You can see the Current Month or Last 6 Months.

```
Log Information
12) View Current Month
13) View Last 6 Months
```

9.5.1 View Current Month

```
Selection: 12

Current Month (October 2021) Logs for ns-cgc00002-a

Oct  4 18:31:06 root: (MHB-CSC)(STANDBY) ns-cgc00002-a is Cluster StandBy - No active tunnels
Oct  4 18:31:11 root: (MHB-CSC)(UP) CSC GRE Cluster was powered ON: Mon  4 Oct 18:31:11 UTC 2021
Oct  4 18:31:15 root: (MHB-CSC)(INFO) Routed Bypass Rules JSON file integrity is OK
Oct  4 18:31:16 root: (MHB-CSC)(INFO) (Index: 0) Rule "0365 Login URLs 1" was created succesfully.
Oct  4 18:31:17 root: (MHB-CSC)(INFO) (Index: 1) Rule "0365 Login URLs 2" was created succesfully.
Oct  4 18:31:18 root: (MHB-CSC)(INFO) (Index: 2) Rule "0365 Login URLs 3" was created succesfully.
Oct  4 18:31:18 root: (MHB-CSC)(INFO) (Index: 3) Rule "portquiz.net" was created succesfully.
Oct  4 18:31:19 root: (MHB-CSC)(INFO) (Index: 4) Rule "0365 Login URLs 4" was created succesfully.
Oct  4 18:31:20 root: (MHB-CSC)(INFO) (Index: 5) Rule "Skype and Teams UDP 1" was created succesfully.
Oct  4 18:31:21 root: (MHB-CSC)(INFO) (Index: 6) Rule "Skype and Teams UDP 2" was created succesfully.
Oct  4 18:31:21 root: (MHB-CSC)(INFO) (Index: 7) Rule "Skype and Teams UDP 3" was created succesfully.
Oct  4 18:32:16 root: (MHB-CSC)(UP) Primary tunnel is active since: Mon  4 Oct 18:32:16 UTC 2021
Oct  4 18:32:16 root: (MHB-CSC)(ACTIVE) ns-cgc00002-a is Cluster Active
Oct  4 18:38:01 cscadmin: (MHB-CSC)(INFO) User 'csccli' was enabled via console.
Oct  4 18:38:07 cscadmin: (MHB-CSC)(INFO) SSH Key file was modified for user 'csccli'.
```

9.5.2 View Last 6 Months

```
Selection: 13

Last 6 Months Logs up to Current Month (October 2021) for ns-cgc00002-a

Oct  4 18:31:06 root: (MHB-CSC)(STANDBY) ns-cgc00002-a is Cluster StandBy - No active tunnels
Oct  4 18:31:11 root: (MHB-CSC)(UP) CSC GRE Cluster was powered ON: Mon  4 Oct 18:31:11 UTC 2021
Oct  4 18:31:15 root: (MHB-CSC)(INFO) Routed Bypass Rules JSON file integrity is OK
Oct  4 18:31:16 root: (MHB-CSC)(INFO) (Index: 0) Rule "0365 Login URLs 1" was created succesfully.
Oct  4 18:31:17 root: (MHB-CSC)(INFO) (Index: 1) Rule "0365 Login URLs 2" was created succesfully.
Oct  4 18:31:18 root: (MHB-CSC)(INFO) (Index: 2) Rule "0365 Login URLs 3" was created succesfully.
Oct  4 18:31:18 root: (MHB-CSC)(INFO) (Index: 3) Rule "portquiz.net" was created succesfully.
Oct  4 18:31:19 root: (MHB-CSC)(INFO) (Index: 4) Rule "0365 Login URLs 4" was created succesfully.
Oct  4 18:31:20 root: (MHB-CSC)(INFO) (Index: 5) Rule "Skype and Teams UDP 1" was created succesfully.
Oct  4 18:31:21 root: (MHB-CSC)(INFO) (Index: 6) Rule "Skype and Teams UDP 2" was created succesfully.
Oct  4 18:31:21 root: (MHB-CSC)(INFO) (Index: 7) Rule "Skype and Teams UDP 3" was created succesfully.
Oct  4 18:32:16 root: (MHB-CSC)(UP) Primary tunnel is active since: Mon  4 Oct 18:32:16 UTC 2021
Oct  4 18:32:16 root: (MHB-CSC)(ACTIVE) ns-cgc00002-a is Cluster Active
```


9.6 Configuration Wizards

In this section, you can run the Configuration Wizard to change GRE Nodes, DNS servers, Etc; Switch tunnels and Update Netskope Nodes Database.

```
Configuration Wizards
14) Change Nodes, DNS servers, Syslog and more.
15) Switch Tunnels - Primary / Secondary.
16) Update Netskope Nodes Databases.
```

9.6.1 Change Nodes, DNS servers, Syslog and more.

With this wizard you can change:

1. Netskope Nodes
2. DNS Servers
3. Bypass Proxy PAC URL
4. Routed Bypass JSON URL
5. Syslog Servers.

```
Selection: 14
Welcome to the CSC GRE Configuration Wizard
Please go to page: Settings -> Security Cloud Platform -> Traffic Steering Section -> GRE and check 'GRE configurations' to validate that 82.68.6.74 is added.
Current Values Configured:
-----
NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74
Primary Tunnel:
    Node : GB,London,LON1
    Node Public IP: 163.116.162.36
    Node Probe: 10.162.6.209
Secondary Tunnel:
    Node : GB,Manchester,MAN1
    Node Public IP: 163.116.165.36
    Node Probe: 10.165.6.209
returnToPrimaryTunnel: true
-----
DNS Servers: 172.19.0.100 ; 1.1.1.1
-----
Bypass Proxy PAC URL
Your current Proxy Bypass PAC URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
-----
Routed Bypass URL
Your current Routed Bypass URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json
-----
Syslog / SIEM information
Primary Syslog / SIEM IP: 172.19.0.199
Secondary Syslog / SIEM IP: Not configured
Syslog / SIEM TCP port: 514
-----
Are you ready to continue?
1) Yes
2) No
Enter your choice: █
```

9.6.1.1 Running the Configuration Wizard

Select Nodes, Auto or Manual.

If you select "Auto" the CSC will select the nearest Netskope Nodes to "GRE Tunnels egress Public IP". If you select Manual, you can choose manually the Nodes Primary and Secondary.

```
-----
Are you ready to continue?
1) Yes
2) No
Enter your choice: 1
-----
NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74

Primary Tunnel:
Node : GB,London,LON1
Node Public IP: 163.116.162.36
Node Probe: 10.162.6.209

Secondary Tunnel:
Node : GB,Manchester,MAN1
Node Public IP: 163.116.165.36
Node Probe: 10.165.6.209

returnToPrimaryTunnel: true

Do you want to change the Netskope Tunnel values?
1) Yes
2) No
Enter your choice: 1
-----
Please, select Manual or Auto Node Selection

1) Manual
2) Auto
3) Quit
Enter your choice: 1
```

Selecting "Manual"

-> Select your Primary Node.

```
-----
Please, select your Primary Node (Country/City/NodeID)

1) AE,Dubai,DXB1      9) BR,SaoPaulo,SAO1      17) DE,Frankfurt,FR1      25) IN,Chennai,MA1      33) NZ,Auckland,AKL1      41) US,LosAngeles,LAX1      49) Not in the list? Input Manually.
2) AE,Dubai,DXB2      10) CA,Montreal,MTL1     18) ES,Madrid,MAD1      26) IN,Delhi,DEL1      34) SE,Stockholm,STO1     42) US,Miami,MIA1      50) Quit
3) AR,BuenosAires,BUE1 11) CA,Toronto,YYZ1      19) FR,Marseille,MRS1   27) IN,Mumbai,BOM1     35) SG,Singapore,SIN1     43) US,NewYork,NYC1
4) AT,Vienna,VIE1      12) CA,Vancouver,YVR1   20) FR,Paris,PAR1      28) IT,Milan,MIL1      36) US,Ashburn,DE1       44) US,Phoenix,PHX1
5) AU,Brisbane,BNE1     13) CH,Zurich,ZUR1       21) GB,London,LON1      29) JP,Osaka,OSA1      37) US,Atlanta,ATL1      45) US,SanFrancisco,SFO1
6) AU,Melbourne,MEL1    14) CL,Santiago,SCL1     22) GB,Manchester,MAN1  30) JP,Tokyo,NRT1      38) US,Chicago,ORD1      46) US,Seattle,SEA1
7) AU,Perth,PER1        15) CO,Bogota,BOG1      23) HK,HongKong,HKG1    31) KR,Seoul,ICN1      39) US,Dallas,DIA1       47) US,Washington,IAD2
8) AU,Sydney,SYD1       16) DE,Dusseldorf,DUS1  24) IL,TelAviv,TLV1     32) NL,Amsterdam,AMS1  40) US,Denver,DEN1       48) ZA,Johannesburg,JNB1

Enter your choice: 43
```

-> Select your Secondary Node

```

Please, select your Secondary Node (Country/City/NodeID)
1) AE,Dubai,DXB1      9) BR,SaoPaulo,SAO1      17) DE,Frankfurt,FRA1      25) IN,Chennai,MAA1      33) NZ,Auckland,AKL1      41) US,LosAngeles,LAX1      49) Not in the list? Input Manually
2) AE,Dubai,DXB2      10) CA,Montreal,YMQ1     18) ES,Madrid,MAD1       26) IN,Delhi,DEL1       34) SE,Stockholm,STO1     42) US,Miami,MIA1       50) Quit
3) AR,BuenosAires,BUE1 11) CA,Toronto,YYZ2      19) FR,Marseille,MRS1    27) IN,Mumbai,BOM1      35) SG,Singapore,SIN1     43) US,NewYork,NYC1
4) AT,Vienne,VIE1     12) CA,Vancouver,YVR1    20) FR,Paris,PAR1        28) IT,Milan,MIL1       36) US,Ashburn,DC11      44) US,Phoenix,PHX1
5) AU,Brisbane,BNE1    13) CH,Zurich,ZUR1       21) GB,London,LON1       29) JP,Osaka,OSA1       37) US,Atlanta,ATL1      45) US,SanFrancisco,SFO1
6) AU,Melbourne,MEL1   14) CL,Santiago,SCL1     22) GB,Manchester,MAN1   30) JP,Tokyo,MTT1       38) US,Chicago,ORD1      46) US,Seattle,SEA1
7) AU,Perth,PER1       15) CO,Bogota,BOG1       23) HK,HongKong,HKG1     31) KR,Seoul,ICN1       39) US,Dallas,DFW1       47) US,Washington,IAD2
8) AU,Sydney,SYD1      16) DE,Dusseldorf,DUS1   24) IL,TelAviv,TLV1      32) NL,Amsterdam,AMS1   40) US,Denver,DEN1       48) ZA,Johannesburg,JNB1
Enter your choice: 47

```

-> Select 'returnToPrimaryTunnel' variable:

Please select 'true' if you want the CSC to return to Primary tunnel (after 10 min of stability) when using Secondary tunnel.

Select 'false' if you want to remain using Secondary Tunnel and not to return to Primary.(Secondary will be nominated as 'new' Primary)

```

'returnToPrimaryTunnel' variable:
Please select 'true' if you want the CSC to return to Primary tunnel (after 10 min of stability) when using Secondary tunnel.
Select 'false' if you want to remain using Secondary Tunnel and not to return to Primary.(Secondary will be nominated as 'new' Primary)
1) true
2) false
Enter your choice: 1

```

DNS Configuration

```

-----
DNS Configuration

Your current DNS Servers are: 172.19.0.100 ; 1.1.1.1

Do you want to change the DNS servers?
1) Yes
2) No
Enter your choice: 2

```

Proxy Bypass Configuration

```

-----
Proxy Bypass Configuration

Your current Proxy Bypass PAC URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac

Do you want to change the Proxy Bypass PAC URL?
1) Yes
2) No
Enter your choice: 2

Do you want to refresh Proxy Bypass List?
1) Yes
2) No
Enter your choice: 2

```


Routed Bypass Configuration

```
Routed Bypass Configuration
Your Routed Bypass URL configured is: https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json
Do you want to change the Routed Bypass URL?
1) Yes
2) No
Enter your choice: 2
Do you want to refresh the Routed Bypass List?
1) Yes
2) No
Enter your choice: 2
```

Syslog / SIEM Configuration

```
Syslog / SIEM Configuration
Primary Syslog / SIEM IP: 172.19.0.199
Secondary Syslog / SIEM IP: Not configured
Syslog / SIEM TCP port: 514
Do you want to change Syslog / SIEM Servers values?
1) Yes
2) No
3) Reset default values
Enter your choice: 2
```

At the end of the Wizard, you will be asked to confirm this values.

```
Please confirm these values:
-----
NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74
Primary Tunnel:
    Node : US,NewYork,NYC1
    Node Public IP: 163.116.135.36
    Node Probe: 10.135.6.209
Secondary Tunnel:
    Node : US,Washington,IAD2
    Node Public IP: 163.116.146.36
    Node Probe: 10.146.6.209
returnToPrimaryTunnel: true
-----
DNS Servers: 172.19.0.100 ; 1.1.1.1
-----
Bypass Proxy PAC URL
Your current Proxy Bypass PAC URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-csc-bypass-pac-documentation.pac
-----
Routed Bypass URL
Your current Routed Bypass URL is https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json
-----
Primary Syslog / SIEM server IP: 172.19.0.199
Syslog / SIEM TCP port IP: 514
-----
Do you want to implement these values? (The CSC will reboot)
1) Yes
2) No
Enter your choice: █
```

Done!

9.6.2 Switch Tunnels - Primary / Secondary.

This Wizard allows to Switch Tunnels Primary to Secondary and vice-versa.

```
Selection: 15

NETSKOPE INFORMATION
GRE tunnels egress Public IP: 82.68.6.74

Primary Tunnel:
    Node : GB,London,LON1
    Node Public IP: 163.116.162.36
    Node Probe: 10.162.6.209
Secondary Tunnel:
    Node : GB,Manchester,MAN1
    Node Public IP: 163.116.165.36
    Node Probe: 10.165.6.209

TUNNEL STATUS
Primary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
Secondary Tunnel (reachability):
    Node Keepalive is: Alive
    GRE Tunnel IP is: Alive
returnToPrimaryTunnel: true

Tunnel Status: Primary tunnel is active since: Fri 8 Oct 03:19:08 BST 2021

HTTP://WWW.NETSKOPE.COM PAGE STATUS
163.116.162.116 London, United Kingdom (LON1)
-----
Do you want to Switch Primary / Secondary Tunnel?
Selecting Yes will disrupt all current connections.

1) Yes
2) No
Enter your choice: ☐
```

9.6.3 Update Netskope Nodes Databases.

This command retrieves the latest Netskope Node Database.

```
Selection: 16

Checking Netskope Nodes Databases...
This CSC has the latest version: 1.2
```

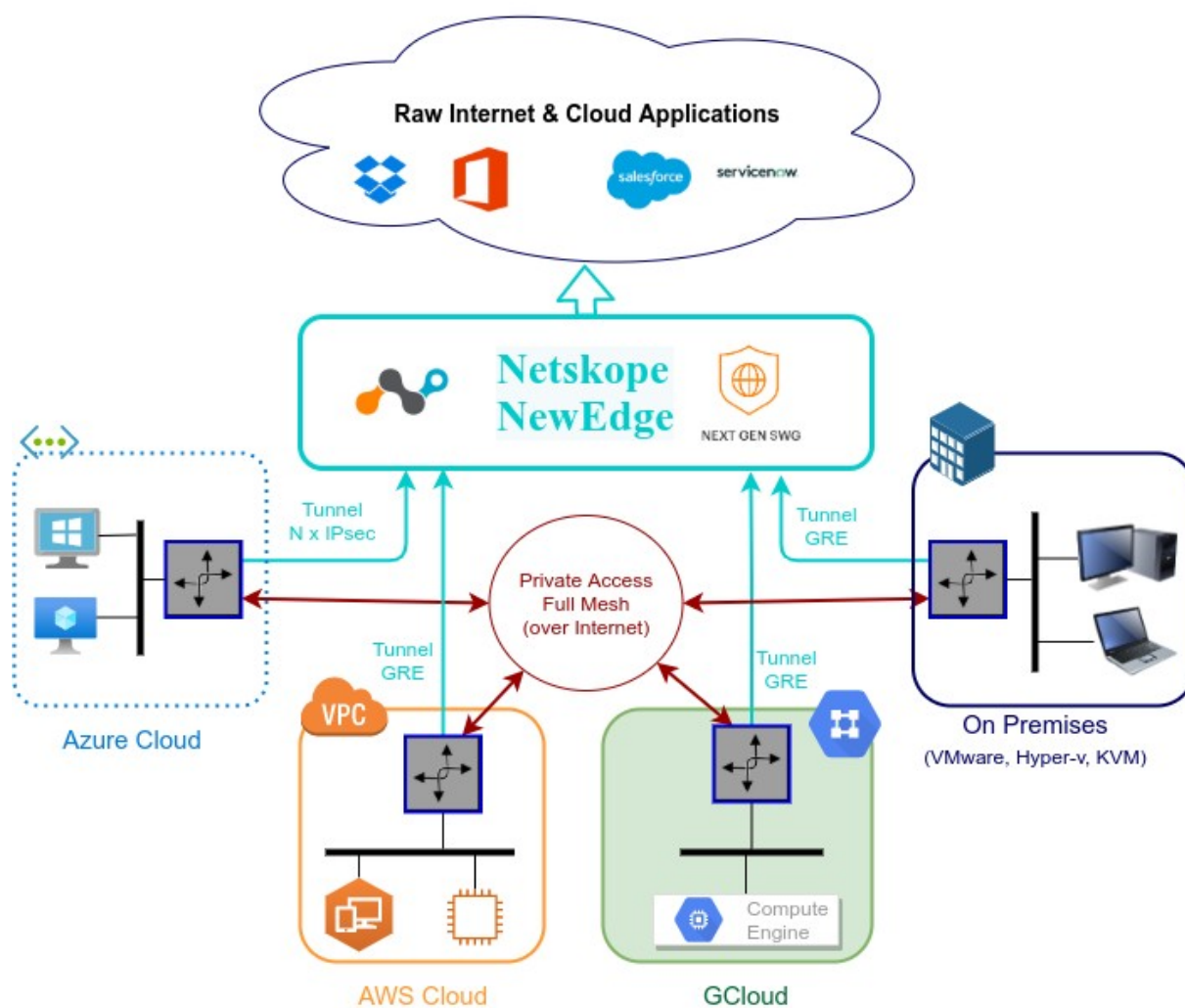
10 Private Cloud Private Access

10.1 What is Private Cloud Private Access (PriCPA)?

Private Cloud Private Access (PriCPA) is a new functionality of the Cloud Security Connector. PriCPA allows you to create a Private Cloud among all CSCs for private traffic. In a matter of minutes, you can build a full mesh encrypted topology between your locations for private traffic with Zero Trust. After making the Private Cloud, you can set up your policies to define who will talk with who inside your Private Cloud.

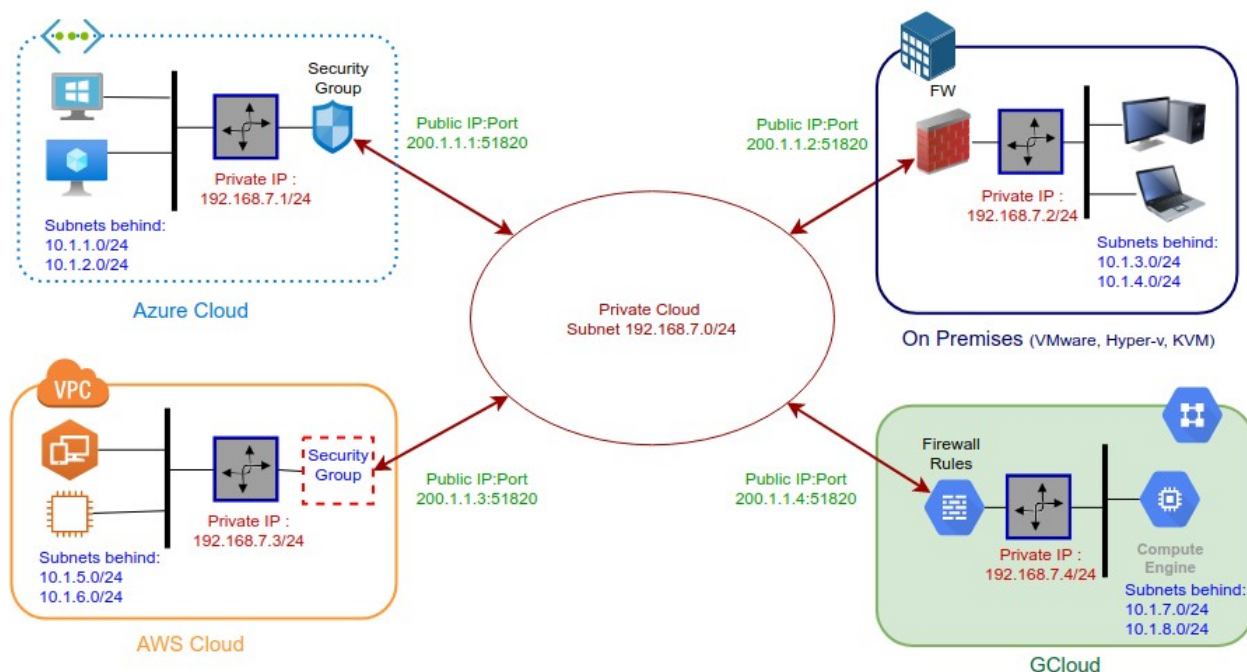
10.2 PriCPA Network Diagrams

10.2.1 High Level Network Diagram



10.2.2 Low Level Network Diagram – PriCPA only

The following network diagram shows the IP addressing for PriCPA.



Steps to design your Private Cloud:

1. Select a Subnet for your Private Cloud. The example above is 192.168.7.0/24. Due to the Subnet is /24, up to 255 CSCs can participate in this Private Cloud.
2. Assign a Cloud Private IP to each CSC. In this example, we are assigning 192.168.7.1 to 192.168.7.4
3. The Public IP to be used will be the same assigned to the Bypass of each CSC. You can choose the UDP port to use at each location. For simplicity, it is recommended to use the same port at all locations.
4. Gather the information of the private Subnets behind each CSC. This information will be required when configuring the Peers.
5. Firewall Rules (or Security Groups Rules): The CSC for Azure, AWS and Gcloud will implement the firewall rules automatically. Manual FW rules are required when the CSC is "On-Premises". The CSC provides a JSON file with the rules required.

10.3 Configuring PriCPA

The Main Menu has a section for Private Access:

```
MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.
```

The configuration of PriCPA is three simple steps:

1. Create the Local Node configuration. This step will initialize and enable Private Access on the Node. The result of this operation will show a "Token" and "Private Access Local JSON file".
2. Initialize the second node in the cluster using the "Token" and "Private Access Local JSON file".
3. Create and distribute the Private Access Peers JSON file to all nodes.

IMPORTANT: We strongly recommend using software with a JSON formatter to create the Peers JSON file, like Visual Code or Notepad ++ . See Appendix C for more detail about how to install these programs and the plugins required.

10.3.1 Create the Local configuration (first node of the cluster)

- From Main Menu, select "18) Configure Private Access."

```
Selection: 18
Private Access Configuration Wizard
Steps to configure Private Access:
- Create Private Access Local Configuration. (This selection also allows to change Local Configuration)
- (optional, if HA is enabled.) Copy Local Configuration to the other CSC in the HA pair.
- Load Private Access Peers JSON configuration file.
1) Create (or change) Private Access Local Configuration
2) Load Private Access Peers JSON configuration file
3) Quit
Enter your choice: 1
```

- Select "1) Create (or change) Private Access Local Configuration"

```

Enter your choice: 1

Private Access is not enabled.

IMPORTANT:
  1) Use 'Manual Configuration' to generate keys and values.
  2) Use 'Token and JSON' to load previous generated values. For example, to configure second CSC on HA Pair.

Do you want to enable Private Access?

1) Manual Configuration
2) Token and JSON
3) Quit
Enter your choice: 1

```

- Select "1) Manual Configuration" and input the values requested.

```

Enter your choice: 1

Before continuing, you need to have the following values ready:
- Node Name. (string)
- (Optional) Location Name. (string)
- (Optional) Description. (string)
- Public IP and UDP Port. (IP:Port)
- Private IP/Subnet of Local Interface. (IP/Subnet Prefix)

Do you want to continue?
1) Yes
2) No
Enter your choice: 1

Please, input the following values:
Node Name (string): ns-cgc00004
(Optional) Location Name (string): MHB-DC-KVM
(Optional) Description (string): CSC GRE Cluster for Netskope located at MHB DC
Public IP and UDP port (IP:Port): 82.68.6.74:51821
Private IP/Subnet of Local Interface (IP/Subnet Prefix): 192.168.7.11/24

Persistent KeepAlive setting:
-> Persistent KeepAlive is required in rare cases:
  a) When the firewall of this site cannot do an outbound NAT without changing the source port.
  b) When incoming connections are not possible at all to this site.

IMPORTANT: We strongly recommend keeping the default value of 'Persistent KeepAlive = no'. Enabling 'Persistent KeepAlive' generates unnecessary traffic and consumes CPU resources.

Do you want to change default value of 'Persistent KeepAlive = no'?
1) Yes
2) No (Recommended)

The values to configure are:
Node Name: ns-cgc00004
Public IP and UDP Port: 82.68.6.74:51821
Private IP/Subnet of Local Interface: 192.168.7.11/24
Location Name: MHB-DC-KVM
Description: CSC GRE Cluster for Netskope located at MHB DC
Persistent KeepAlive: no

Do you want to apply this values?
1) Yes
2) No
Enter your choice: 1

```

- Apply values


```
Enter your choice: 1
Private Access - Private Access service is enabled on ns-cgc00004-a. Private Access is enabled.
Please, copy the following values in a safe place to configure the other CSC on the High Availability Pair. Discard this message if you are doing a single deployment

Token: RULYd21DZ1B0NnRjT1I4SWNVVkdWbEzBSHkrdEZEYzdMT29KRKhLbFpFbz0K
Private Access Local Config JSON file:
{
  "peers": [
    {
      "nodeName": "ns-cgc00004",
      "location": "MHB-DC-KVM",
      "description": "CSC GRE Cluster for Netskope located at MHB DC",
      "publicKey": "6fR2Hy30AXWlzM/+cMqVZ5FLhj9pwTvLnGq602S1FwM=",
      "publicIpAndUdpPort": "82.68.6.74:51821",
      "privateCidrIp": "192.168.7.11/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
}
```

IMPORTANT: Keep this values on a safe place.

IMPORTANT: The "Token" and "Private Access Local Config JSON file" will be used to create the local configuration on the second node of the CSC GRE Cluster. Please, keep these values in a safe place. You can use these values to reconfigure any node of the CSC GRE Cluster if necessary in the future. For example, if you want to change the IPs or descriptions.

10.3.2 Create the Local configuration (second node of the cluster)

SSH the second node if the Cluster and input the "Token" and "Private Access Local Config JSON file".

```
MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.
e) Exit
Selection: 18
Private Access Configuration Wizard
Steps to configure Private Access:
- Create Private Access Local Configuration. (This selection also allows to change Local Configuration)
- (optional, if HA is enabled.) Copy Local Configuration to the other CSC in the HA pair.
- Load Private Access Peers JSON configuration file.
1) Create (or change) Private Access Local Configuration
2) Load Private Access Peers JSON configuration file
3) Quit
Enter your choice: 1 2
Private Access is not enabled.
IMPORTANT:
1) Use 'Manual Configuration' to generate keys and values.
2) Use 'Token and JSON' to load previous generated values. For example, to configure second CSC on HA Pair.
Do you want to enable Private Access?
1) Manual Configuration
2) Token and JSON
3) Quit
Enter your choice: 2 3
Before continuing, you need to have ready the values generated on the First CSC on the HA Pair.:
1 - Token (string)
2 - Private Access Local Config JSON file. (JSON File) Info required
Do you want to continue?
1) Yes
2) No
Enter your choice: 1 4
```

```
Do you want to continue?
1) Yes
2) No
Enter your choice: 1 1
Please, input the following values:
Token (string): RULYd21DZ1B0NnRjT1I4SWNVVkdwbEZBSHkrdEZEYzdMT29kRkhlbFpFbz0K 2
Please, paste 'Private Access Local Config JSON file' and press 'Enter' if required.
NOTE: If the json file has errors, it is possible that the script will hang. Press '}' and 'Enter' to end the operation.
Private Access Local Config JSON file: {
  "peers": [
    {
      "nodeName": "ns-cgc00004",
      "location": "MHB-DC-KVM",
      "description": "CSC GRE Cluster for Netskope located at MHB DC",
      "publicKey": "6fR2Hy30AXWizM/+cMqVZSFLhj9pwTivLnGq602SlFwM=",
      "publicIpAndUdpPort": "82.68.6.74:51821",
      "privateCirdIp": "192.168.7.11/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
} 3
Private Access Local Config JSON file imported successfully

The values to configure are:
Node Name: "ns-cgc00004"
Public IP and UDP Port: 82.68.6.74:51821
Private IP/Subnet of Local Interface: 192.168.7.11/24
Location Name: "MHB-DC-KVM"
Description: "CSC GRE Cluster for Netskope located at MHB DC"
Persistent KeepAlive: no
Confirm Values

Do you want to apply this values?
1) Yes
2) No
Enter your choice: 1 4
Private Access - Private Access service is enabled on ns-cgc00004-b. Private Access is enabled.
```

10.3.3 Create the Private Access Peers JSON file

The Private Access Peers JSON file contains:

1. The Local configuration of each Peer.
2. The "networks" behind each Peer.
3. The "privateApps" allowed to be reached on each Peer.

Here some examples.

10.3.3.1 Full mesh Private Access Peers JSON file

Consider the following example:

We have 3 nodes and we want to allow full communication between sites for all port and protocols.

The Local Config JSON file of each node is:

ns-cgc00001

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00001",
      "description": "Node on VMware Server 1",
      "location": "HQ",
      "publicKey": "yAnz5TF+lXXJte14tji3zIMNq+hd2rYUlgJBgB3fBmk=",
      "publicIpAndUdpPort": "200.1.1.1:51821",
      "privateCirdIp": "192.168.7.1/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
}
```

ns-cgc00002

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00002",
      "description": "Node on VMware Server 2",
      "location": "Datacentre 2",
      "publicKey": "xTIBA5rboUvnH4htodjb6e697QjLERT1NAB4mZqp8Dg=",
      "publicIpAndUdpPort": "200.1.1.2:51821",
      "privateCirdIp": "192.168.7.2/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
}
```

ns-cgc00003

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00003",
      "description": "Node on VMware Server 3",
      "location": "Branch",
      "publicKey": "TrMvSoP4jYQjY6RIzBgbssQqY3vxl2Pi+y71IOWWXX0=",
      "publicIpAndUdpPort": "200.1.1.3:51821",
      "privateCirdIp": "192.168.7.3/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
}
```


Firstly, we need to create our "basic" Peers Configuration JSON file: It contains the Local Configuration of each Node plus the "networks" behind each node.

Basic Peers Configuration JSON file

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00001",
      "description": "Node on VMware Server 1",
      "location": "HQ",
      "publicKey": "yAnz5TF+IXXJte14tji3zlMNq+hd2rYUlgJBgB3fBmk=",
      "publicIpAndUdpPort": "200.1.1.1:51821",
      "privateCirdIp": "192.168.7.1/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.1.1.0/24",
        "10.1.2.0/24"
      ],
      "privateApps": []
    },
    {
      "nodeName": "ns-cgc00002",
      "description": "Node on VMware Server 2",
      "location": "Datacentre 2",
      "publicKey": "xTIBA5rboUvnH4htodjb6e697QjLERT1NAB4mZqp8Dg=",
      "publicIpAndUdpPort": "200.1.1.2:51821",
      "privateCirdIp": "192.168.7.2/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.2.1.0/24",
        "10.2.2.0/24"
      ],
      "privateApps": []
    },
    {
      "nodeName": "ns-cgc00003",
      "description": "Node on VMware Server 3",
      "location": "Branch",
      "publicKey": "TrMvSoP4jYQlY6RlzbgbssQqY3vxl2Pi+y71lOWWXX0=",
      "publicIpAndUdpPort": "200.1.1.3:51821",
      "privateCirdIp": "192.168.7.3/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.3.1.0/24",
        "10.3.2.0/24"
      ],
      "privateApps": []
    }
  ]
}
```

In this "Basic Peers Configuration JSON file" we have:

- **Green:** The Local values generated at each node.
- **Yellow:** The Subnets behind each node
- **Red:** Nothing. No private Apps configured.

If you deployed this "Basic Peers Configuration JSON file" to all CSCs, you have created the Private Cloud. All Peers will be visible to each other, but no traffic between subnets will be allowed because there is no "privateApps" configured.

If we want to allow traffic any to any between subnets, we need to add the corresponding "privateApps" to each node. For example for node: "ns-cgc00001"

```
ns-cgc00001
{
  "nodeName": "ns-cgc00001",
  "description": "Node on VMware Server 1",
  "location": "HQ",
  "publicKey": "yAnz5TF+IXXJte14tji3zIMNq+hd2rYUlgJBgB3fBmk=",
  "publicIpAndUdpPort": "200.1.1.1:51821",
  "privateCirdIp": "192.168.7.1/24",
  "persistentKeepAlive": "no",
  "networks": [
    "10.1.1.0/24",
    "10.1.2.0/24"
  ],
  "privateApps": [
    {
      "description": "Allow all traffic to this site",
      "ipProtocol": "all",
      "sourceCirdIp": [
        "0.0.0.0/0"
      ],
      "destinationCirdIp": [
        "10.1.1.0/24",
        "10.1.2.0/24"
      ],
      "destinationSinglePorts": [
        ""
      ],
      "destinationPortRange": {
        "fromPort": "",
        "toPort": ""
      }
    }
  ]
}
```

In this case, we added a "privateApp" that allows any source IPs (0.0.0.0/0) to reach the "networks" (10.1.1.0/24 and 10.1.2.0/24) using "all" protocols ("ipProtocol" : "all".)

Now, completing our "Peers Configuration JSON file":

Full Mesh Peers Configuration JSON file.

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00001",
      "description": "Node on VMware Server 1",
      "location": "HQ",
      "publicKey": "yAnz5TF+IXXJte14tj3zlMNq+hd2rYUlgJBgB3fBmk=",
      "publicIpAndUdpPort": "200.1.1.1:51821",
      "privateCirdIp": "192.168.7.1/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.1.1.0/24",
        "10.1.2.0/24"
      ],
      "privateApps": [
        {
          "description": "Allow all traffic to this site",
          "ipProtocol": "all",
          "sourceCirdIp": [
            "0.0.0.0/0"
          ],
          "destinationCirdIp": [
            "10.1.1.0/24",
            "10.1.2.0/24"
          ],
          "destinationSinglePorts": [
            ""
          ],
          "destinationPortRange": {
            "fromPort": "",
            "toPort": ""
          }
        }
      ]
    },
    {
      "nodeName": "ns-cgc00002",
      "description": "Node on VMware Server 2",
      "location": "Datacentre 2",
      "publicKey": "xTlBASrboUvnH4htodjb6e697QJlERt1NAB4mZqp8Dg=",
      "publicIpAndUdpPort": "200.1.1.2:51821",
      "privateCirdIp": "192.168.7.2/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.2.1.0/24",
        "10.2.2.0/24"
      ],
      "privateApps": [
        {
          "description": "Allow all traffic to this site",
          "ipProtocol": "all",
          "sourceCirdIp": [
            "0.0.0.0/0"
          ],
          "destinationCirdIp": [
            "10.2.1.0/24",
            "10.2.2.0/24"
          ],
          "destinationSinglePorts": [
            ""
          ],
          "destinationPortRange": {
            "fromPort": "",
            "toPort": ""
          }
        }
      ]
    },
    {
      "nodeName": "ns-cgc00003",
      "description": "Node on VMware Server 3",
      "location": "Branch",
      "publicKey": "TrMvSoP4jYQlY6RlZBgbsQqY3vxl2Pi+y71IOWWXX0=",
      "publicIpAndUdpPort": "200.1.1.3:51821",
      "privateCirdIp": "192.168.7.3/24",
      "persistentKeepAlive": "no",
      "networks": [
        "10.3.1.0/24",
        "10.3.2.0/24"
      ],
      "privateApps": [

```



```

{
  "description": "Allow all traffic to this site",
  "ipProtocol": "all",
  "sourceCirdIp": [
    "0.0.0.0/0"
  ],
  "destinationCirdIp": [
    "10.3.1.0/24",
    "10.3.2.0/24"
  ],
  "destinationSinglePorts": [
    ""
  ],
  "destinationPortRange": {
    "fromPort": "",
    "toPort": ""
  }
}
]
}

```

Done! Your task is to implement this JSON file on all CSCs and you will have full connectivity any to any for all protocols.

10.3.3.2 Understanding "privateApps" configuration and values

Question 1: Where to configure the "privateApps"?

Only on the node that has the "destinationCirdIp": [], that belongs to its "networks".

Example: I want to allow access to "destinationCirdIp": ["10.1.1.50/32"]. The rule must be created on node ns-cgc00001 that has "networks": ["10.1.1.0/24", "10.1.2.0/24"]

Question 2 : What about the values to configure?

On "privateApps" section there are two types of values to input:

Accepts single value only -> ""

Accepts single or multiple values -> []

```

"privateApps": [
  {
    "description": "",
    "ipProtocol": "",
    "sourceCirdIp": [],
    "destinationCirdIp": [],
    "destinationSinglePorts": [],
    "destinationPortRange": {
      "fromPort": "",
      "toPort": ""
    }
  }
]

```

Examples:

Single value (""):

```
"description": " Intranet Servers",  
"ipProtocol": "tcp",
```

Single or Multiple values ([]):

```
"sourceCirdIp": ["0.0.0.0/0"],  
"destinationCirdIp": ["10.1.1.100/32", "10.1.2.100/32"],  
"destinationSinglePorts": [ "80", "443" ],
```

The following table shows all field and values accepted

Field	Value Type	Values to configure	Example
"description": "",	Single	String	"description": "Intranet Server Access",
"ipProtocol": "",	Single	tcp,udp,icmp or all	"ipProtocol": "tcp",
"sourceCirdIp": [],	Single or Multiple	Networks in the range of: 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 and 0.0.0.0/0	"sourceCirdIp": ["10.2.1.0/24", "10.2.2.0/24", "10.3.1.0/24", "10.3.2.0/24"],
"destinationCirdIp": [],	Single or Multiple	Networks in the range of ¹¹ : 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16	"destinationCirdIp": ["10.1.1.100/32", "10.1.1.200/32"],
"destinationSinglePorts": [],	Single or Multiple	Single Port of the range 1 to 65535	"destinationSinglePorts": ["80", "443"],
"destinationPortRange": { "fromPort": "", "toPort": "" }	Single	Single Port of the range 1 to 65535	"destinationPortRange": { "fromPort": "3780", "toPort": "3784" }

¹¹ The expected value here is a value that belongs to the "network" defined behind the CSC. For example, of the network behind the CSC is 10.1.1.0/24, any destination configured must belong to 10.1.1.0/24, like 10.1.1.100/32.

10.3.3.3 Example of "privateApps" for a Windows Domain controller

The following example shows how to create rules to allow access to your Domains Controllers.

The port information was taken from this article:

<https://docs.microsoft.com/en-us/troubleshoot/windows-server/identity/config-firewall-for-ad-domains-and-trusts>

Example: Domain Controllers IPs: "10.2.1.100/32" and "10.2.2.100/32" on Node ns-cgc00002 of previous example

```
"privateApps": [
  {
    "description": "Domain Controllers TCP",
    "ipProtocol": "tcp",
    "sourceCirdIp": [ "0.0.0.0/0" ],
    "destinationCirdIp": [ "10.2.1.100/32", "10.2.2.100/32" ],
    "destinationSinglePorts": [ "135", "464", "389", "636", "3268", "3269", "53", "88", "445" ],
    "destinationPortRange": { "fromPort": "49152", "toPort": "65535" }
  },
  {
    "description": "Domain Controllers UDP",
    "ipProtocol": "udp",
    "sourceCirdIp": [ "0.0.0.0/0" ],
    "destinationCirdIp": [ "10.2.1.100/32", "10.2.2.100/32" ],
    "destinationSinglePorts": [ "123", "464", "389", "53", "88" ],
    "destinationPortRange": { "fromPort": "", "toPort": "" }
  },
  {
    "description": "Domain Controllers Ping",
    "ipProtocol": "icmp",
    "sourceCirdIp": [ "0.0.0.0/0" ],
    "destinationCirdIp": [ "10.2.1.100/32", "10.2.2.100/32" ],
    "destinationSinglePorts": [],
    "destinationPortRange": { "fromPort": "", "toPort": "" }
  }
]
```

10.3.3.4 Example of "privateApps" for Internal Web Server.

In this example, we are showing how to configure access to users on ns-cgc00001 to an Internal Web server located behind node Node ns-cgc00003.

Example: Web Server "10.3.1.200/32" on Node ns-cgc00003 of previous example. Allow access to all users behind ns-cgc00001

```
"privateApps": [
  {
    "description": "Web Server 3",
    "ipProtocol": "tcp",
    "sourceCirdIp": [ "10.1.1.0/24", "10.1.2.0/24" ],
    "destinationCirdIp": [ "10.3.1.200/32" ],
    "destinationSinglePorts": [ "80", "443" ],
    "destinationPortRange": { "fromPort": "", "toPort": "" }
  }
]
```


10.3.4 Load the "Private Access Peers JSON file" to the CSCs.

After the Local Configuration is done and the "Private Access Peers JSON file" is created, the next task is to distribute and apply it on each CSC.

There are three methods available:

1. URL: (Recommended) Using "Private Access Peers URL" and running the command "Refresh Private Access Peers URL" using AWS Systems Manager or Rundeck.
2. DevOps: Distribute the JSON file on all CSC and run the command "Reload Private Access Peers URL" using AWS Systems Manager or Rundeck.
3. Manual: Copy/Paste the JSON file on each CSCs.

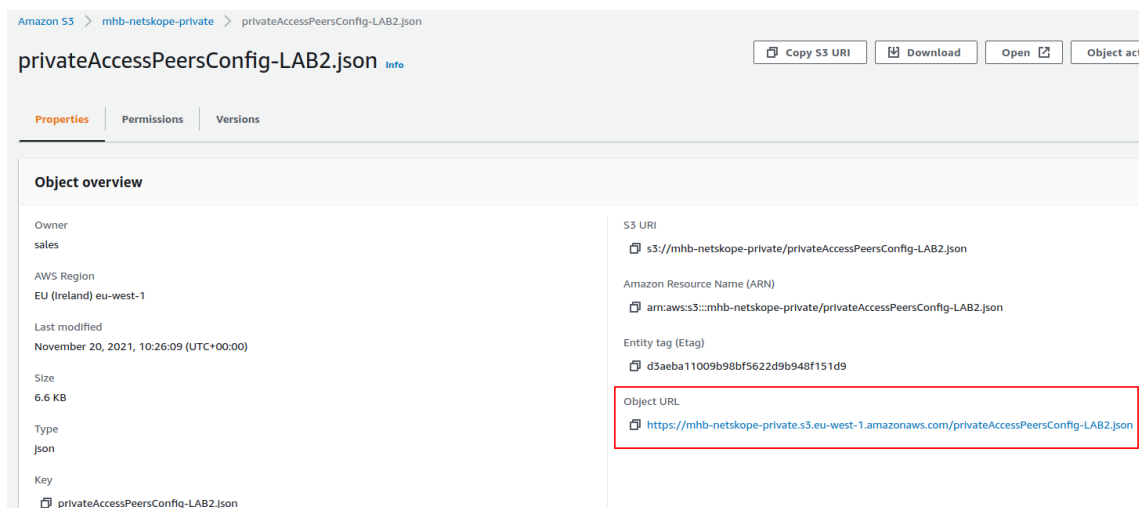
In this section we are going to explain two methods: URL and Manual Copy. The DevOps method is explained on Section 12: DevOps operations.

10.3.4.1 Using "Private Access Peers URL"

This is the recommended method. The steps to configure are:

1. Place the Private Access Peers JSON file on an internal web server or an AWS bucket¹² or similar. Obtain the download URL.

Example of AWS bucket:



2. Configure the URL on each CSC.

Ssh the each CSC and go to Main Menu -> 18) Configure Private Access

¹² See Appendix D to learn how to secure an AWS S3 bucket by Source IP.

```

MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.

e) Exit

Selection: 18 1

Private Access Configuration Wizard

Steps to configure Private Access:

- Create Private Access Local Configuration. (This selection also allows to change Local Configuration)
- (optional, if HA is enabled.) Copy Local Configuration to the other CSC in the HA pair.
- Load Private Access Peers JSON configuration file.

1) Create (or change) Private Access Local Configuration
2) Load Private Access Peers JSON configuration file
3) Quit
Enter your choice: 2 2

Private Access is enabled.

Please, Select Method:
1) Private Access Peers URL
2) Manual (Paste Private Access Peers JSON File)
3) Reset to Default Values
4) Quit
Enter your choice: 1 3

*** Private Access Peers URL is not configured ***

Do you want to configure the Private Access Peers URL?
1) Yes
2) No
Enter your choice: 1 4

Please, input Private Access Peers URL
Private Access Peers URL: https://mhb-netskope-private.s3.eu-west-1.amazonaws.com/privateAccessPeersConfig-LAB2.json

Do you want to refresh the Private Access Peers List?
1) Yes
2) No
Enter your choice: 1 6

Private Access Peers JSON file imported successfully

```

At this moment, you have the option to review the privateApps to configure in Compact or JSON format and to apply the values.

```

Private Access Peers JSON file imported successfully

You can review your values before to apply. Please, Select 'Compact' or 'Json' format.
1) Compact
2) Json
3) No review is needed
Enter your choice: 1

Private Apps:
Index: 0, NodeName: ns-cgc00004, Location: MHB-DC-KVM, publicIpAndUdpPort: 82.68.6.74:51821, privateCirdIp: 192.168.7.11/24, Private Apps Qty: 7
Index: 1, NodeName: ns-cgc00005, Location: MHB-DC-KVM, publicIpAndUdpPort: 82.68.6.76:51820, privateCirdIp: 192.168.7.21/24, Private Apps Qty: 0
Index: 2, NodeName: ns-cgc00006, Location: MHB-BH-DC, publicIpAndUdpPort: 217.155.196.81:51820, privateCirdIp: 192.168.7.20/24, Private Apps Qty: 1

Do you want to apply this values?
1) Yes
2) No
Enter your choice: 1

Creating Private Apps
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Intranet Server' was created successfully. (destinationSinglePorts)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Domain Controllers TCP' was created successfully. (destinationSinglePorts)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Domain Controllers TCP' was created successfully. (destinationPortRange)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Domain Controllers UDP' was created successfully. (destinationSinglePorts)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Domain Controllers PING' was created successfully.
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Syslog server' was created successfully. (destinationSinglePorts)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'ICMP to 172.19.0.133' was created successfully.
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'All protocol 192.168.6.0/24' was created successfully.
Private Access - (Index: 2, Node: ns-cgc00006) Private App 'BH - SSH and RDP' was created successfully. (destinationSinglePorts)

Adding Peers:
Private Access - Node: ns-cgc00005 added successfully.
Private Access - Node: ns-cgc00006 added successfully.

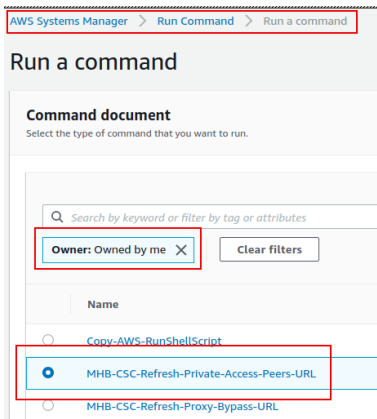
Private Access - Private Access Peers List updated successfully.

```

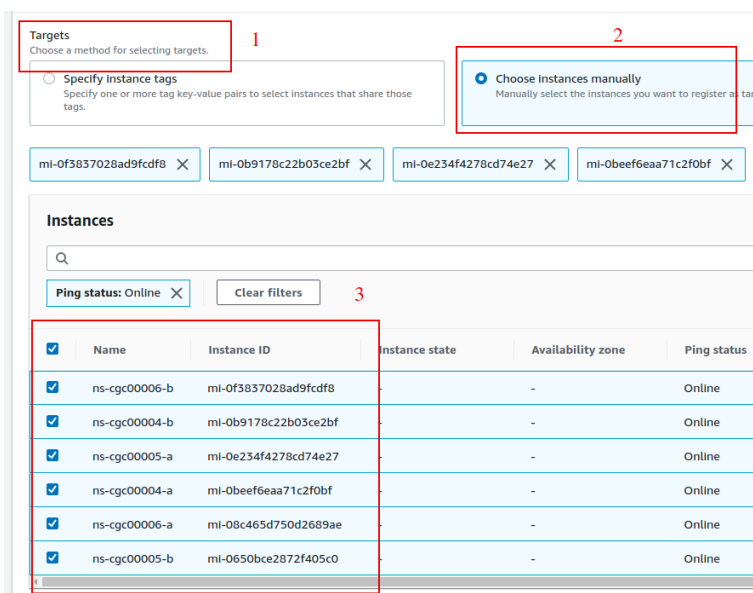
3. The next time you want to refresh the Private Access Peers JSON file, update the file, deploy it on the same location URL and Run Command: "Refresh Private Access Peers URL" using AWS SSM Agent or Rundeck.

AWS System Manager:

- Go to AWS Systems Manager -> Run Command -> and Select "MHB-CSC-Refresh-Private-Access-Peers-URL"



- Move down the screen and select all CSCs:



- Go to the bottom of the page and click "Run". The next page shows the status of the command on each CSC.

Command ID: e7c8bfa2-e045-4df0-8216-4721be8d4249 was successfully sent!

AWS Systems Manager > Run Command > Command ID: e7c8bfa2-e045-4df0-8216-4721be8d4249

Command ID: e7c8bfa2-e045-4df0-8216-4721be8d4249

Command status

Overall status ✔ Success	Detailed status ✔ Success	# targets 6	# completed 6
-----------------------------	------------------------------	----------------	------------------

Targets and outputs

Q

	Instance ID	Instance name	Status	Detailed Status
<input type="radio"/>	mi-0650bce2872f405c0	ns-cgc00005-b	✔ Success	✔ Success
<input type="radio"/>	mi-08c465d750d2689ae	ns-cgc00006-a	✔ Success	✔ Success
<input type="radio"/>	mi-0beef6eaa71c2f0bf	ns-cgc00004-a	✔ Success	✔ Success
<input type="radio"/>	mi-0e234f4278cd74e27	ns-cgc00005-a	✔ Success	✔ Success
<input type="radio"/>	mi-0b9178c22b03ce2bf	ns-cgc00004-b	✔ Success	✔ Success
<input type="radio"/>	mi-0f3837028ad9fcd8	ns-cgc00006-b	✔ Success	✔ Success

- To see the individual result, right click on the Instance ID and open it on a new TAB. Check the "Output"

AWS Systems Manager > Run Command > Command ID: e7c8bfa2-e045-4df0-8216-4721be8d4249 > Output on: mi-0650bce2872f405c0

Output on mi-0650bce2872f405c0

Step 1 - Command description and status

Status ✔ Success	Detailed status ✔ Success
Step name Runscripts	Start time Sat, 20 Nov 2021 22:39:33 GMT

▼ Output

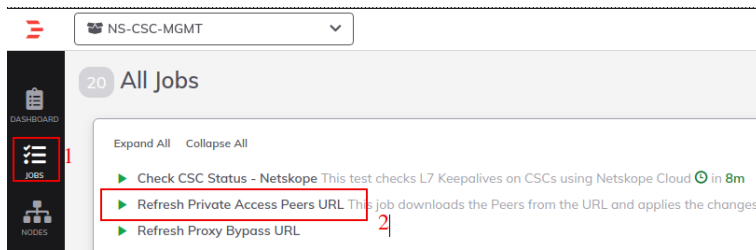
The command output displays a maximum of 48,000 characters. You can view the complete command output in either Amazon S3 or CloudWatch Logs, if:

```
Private Access - Private Access Peers JSON file imported successfully.

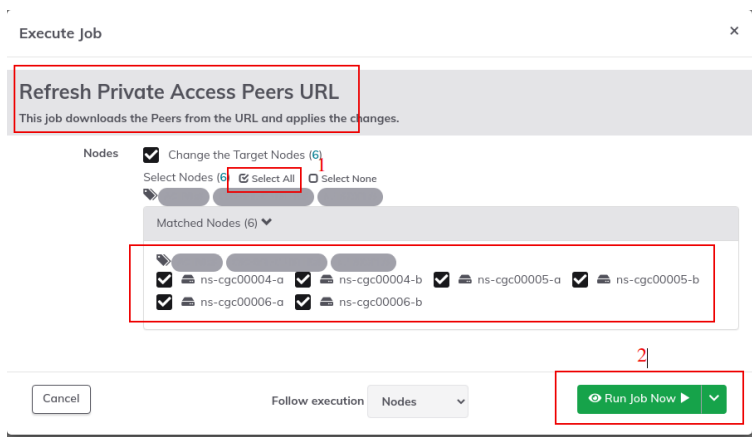
Creating Private Apps
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Intranet Server' was created successfully.
(destinationSinglePorts)
Private Access - (Index: 0, Node: ns-cgc00004) Private App 'Domain Controllers TCP' was created successfully.
(destinationSinglePorts)
```

Using Rundeck

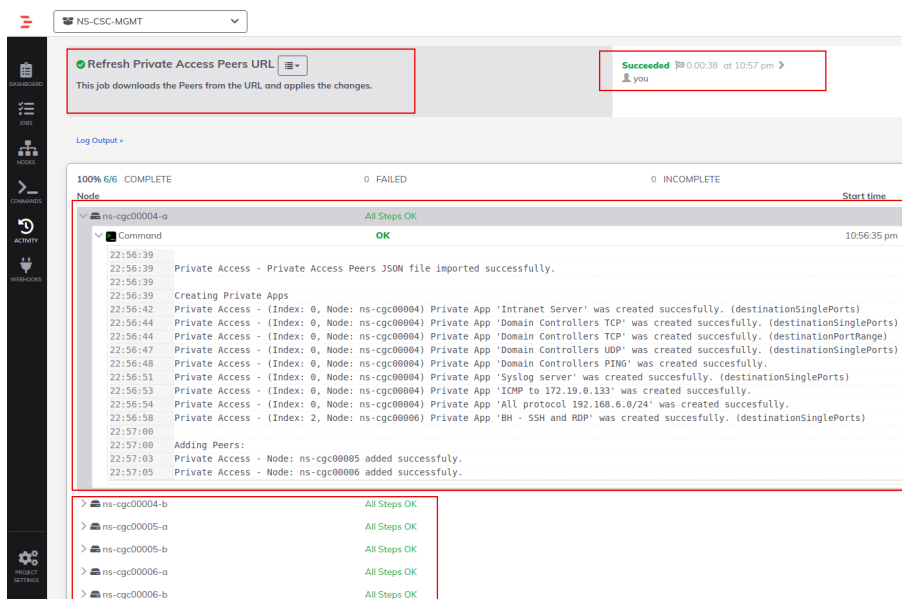
- Go to the Project <name> -> All Jobs -> Run " Refresh Private Access Peers URL"



- Select ALL nodes and click Run.



- Wait to succeeded. You can click on "command" to see the results node by node.



10.4 Show Configurations and Status Private Access.

10.4.1 Via SSH console

From Main Menu, go to 17) Show Configurations and Status Private Access.

```
MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.

e) Exit

Selection: 17

Show Configuration and Status Private Access

Please, select an option:

1) Show Peer/s Status
2) Show Peers Json file (active)
3) Show Local Configuration
4) Show Firewall Local Rules
5) Quit
Enter your choice: █
```

10.4.1.1 Show Peer/s Status

In this menu you can see "All Peers Status" or by peer "Select Peer".

```
Enter your choice: 1

Please, select an option:

1) Show ALL Peers Status
2) Select Peer
3) Quit
Enter your choice: █
```

1. Show All Peers Status

```
1) Show ALL Peers Status
2) Select Peer
3) Quit
Enter your choice: 1

Peer 'ns-cgc00005' -> 192.168.7.21 is Alive. Source Port OK. Using '51820'
Peer 'ns-cgc00006' -> 192.168.7.20 is Alive. Source Port OK. Using '51820'
```

IMPORTANT: This section show is the Peer is Alive and the "Source Port" that arrives at this node from the Peer. The Source Port information is essential to validate that the NAT on the Remote Peer is correct or if the FW on the other end is changing the Source Port. Please, correct the NAT on the remote Peer if you see that the Source Port differs from the expected.

2. Select Peer

This section shows a more detailed information about the Peer.

```
Please, select a Peer
1) "ns-cgc00005"
2) "ns-cgc00006"
3) Quit
Enter your choice: 2

Peer Status:
Peer "ns-cgc00006" -> 192.168.7.20 is Alive. Source Port OK. Using '51820'

Peer Counters:
Latest Communication: Sun 21 Nov 09:17:39 UTC 2021
Transfer: 5.9Mi received, 108Mi sent

Peer Configuration:
{
  "nodeName": "ns-cgc00006",
  "description": "CSC on Bournemouth branch",
  "location": "MHB-BH-DC",
  "publicKey": "B00gLRseH+p3tWgk04j9rVawX2Fbqkj0d0JLyM1TsmI=",
  "publicIpAndUdpPort": "217.155.196.81:51820",
  "privateCirdIp": "192.168.7.20/24",
  "persistentKeepAlive": "no",
  "networks": [
    "10.151.1.0/24"
  ],
  "privateApps": [
    {
      "description": "BH - SSH and RDP",
      "ipProtocol": "tcp",
      "sourceCirdIp": [
        "172.19.0.0/24"
      ],
      "destinationCirdIp": [
        "10.151.1.4/32",
        "10.151.1.5/32",
        "10.151.1.6/32"
      ],
      "destinationSinglePorts": [
        "22",
        "3389"
      ],
      "destinationPortRange": {
        "fromPort": "",
        "toPort": ""
      }
    }
  ]
}
```

10.4.1.2 Show Peers Json file (active)

This menu shows the active Private Access Peers Json file.

```
Selection: 17
Show Configuration and Status Private Access
Please, select an option:
1) Show Peer/s Status
2) Show Peers Json file (active)
3) Show Local Configuration
4) Show Firewall Local Rules
5) Quit
Enter your choice: 2

{
  "peers": [
    {
      "nodeName": "ns-cgc00004",
      "description": "Node located at MHB DC",
      "location": "MHB-DC-KVM",

```

10.4.1.3 Show Local Configuration

This menu shows the Local configuration of the node.

```
Selection: 17
Show Configuration and Status Private Access
Please, select an option:
1) Show Peer/s Status
2) Show Peers Json file (active)
3) Show Local Configuration
4) Show Firewall Local Rules
5) Quit
Enter your choice: 3

The Local Configuration menu shows the initial configuration of the node. Private Apps and Networks are not shown here. Check 'Show Peers Json file' to see all information.

{
  "peers": [
    {
      "nodeName": "ns-cgc00004",
      "description": "CSC GRE Cluster for Netskope located at MHB DC",
      "location": "MHB-DC-KVM",
      "publicKey": "N5s1sRUmrACXJCR3XqZz4gaekarPMT+CgbJyS0/OBM=",
      "publicIpAndUdpPort": "82.68.6.74:51821",
      "privateCidrIp": "192.168.7.11/24",
      "persistentKeepAlive": "no",
      "networks": [],
      "privateApps": []
    }
  ]
}

Press Enter to continue...
```

10.4.1.4 Show Firewall Local Rules

This menu shows in JSON format the Rules required on your local FW for this node.

```
Selection: 17
Show Configuration and Status Private Access
Please, select an option:
1) Show Peer/s Status
2) Show Peers Json file (active)
3) Show Local Configuration
4) Show Firewall Local Rules
5) Quit
Enter your choice: 4

This JSON File shows the required Inbound and Outbound Firewall Rules for the CSC's 'localPrivateIp'.

{
  "nodeName": "ns-cgc00004",
  "localPrivateIp": "192.168.1.61",
  "inboundFirewallRules": [
    {
      "localUdpPort": "51821",
      "peersPublicSourceIP": [
        "82.68.6.76",
        "217.155.196.81"
      ]
    }
  ],
  "outboundFirewallRules": [
    {
      "remoteUdpPort": "51820",
      "peersPublicDestinationIP": [
        "82.68.6.76",
        "217.155.196.81"
      ]
    }
  ]
}

Press Enter to continue...
```


10.4.2 Via AWS Systems Manager or Rundeck

In this case, the information provided is only "Show ALL Peer Status"

10.4.2.1 AWS Systems Manager

Go to AWS Systems Manager and Run Command: "MHB-CSC-Show-Private-Access-ALL-Peers-Status" and select the Nodes. The result will show:

The screenshot shows the AWS Systems Manager console interface. At the top, it indicates the command ID and the output on instance 'mi-08c465d750d2689ae'. The main section is titled 'Output on mi-08c465d750d2689ae'. Under 'Step 1 - Command description and status', the status is 'Success' and the start time is 'Sun, 21 Nov 2021 09:46:15 GMT'. The 'Output' section shows the command output, which includes two lines of peer status information: 'Peer 'ns-cgc00004' -> 192.168.7.11 is Alive. Source Port OK. Using '51821'' and 'Peer 'ns-cgc00005' -> 192.168.7.21 is Alive. Source Port OK. Using '51820''.

10.4.2.2 Rundeck

On Rundeck, run Job: "Show Private Access ALL Peers Status". Select the nodes. The output will show:

The screenshot shows the Rundeck web interface. At the top, the job 'Show Private Access ALL Peers Status' is shown as 'Succeeded' with a 'you' icon. Below this, the 'Log Output' section shows the job progress. The job is '100% 6/6 COMPLETE' with '0 FAILED'. The 'Node' list shows six nodes: 'ns-cgc00004-a', 'ns-cgc00004-b', 'ns-cgc00005-a', 'ns-cgc00005-b', 'ns-cgc00006-a', and 'ns-cgc00006-b'. Each node has a status of 'All Steps OK'. The 'Command' section for 'ns-cgc00004-a' shows the output: '09:50:20', '09:50:21 Peer 'ns-cgc00005' -> 192.168.7.21 is Alive. Source Port OK. Using '51820'', and '09:50:22 Peer 'ns-cgc00006' -> 192.168.7.20 is Alive. Source Port OK. Using '51820''.

10.5 Configure CSC Remote Management via Private Access.

When the CSC is in HA, like the CSC GRE Cluster, only the active node belongs to the Private Cloud. The Standby is not. For this reason, if you want to reach the Standby node using SSH, you must configure Remote Management on both CSC of the Cluster (or HA pair).

The configuration is via SSH Main Menu. You need to add the "Management Networks". For example, in your primary Datacentre, you have the Subnet 172.25.0.0/24, and from that Subnet, you want to reach ALL CSCs on the Private Cloud.

The configuration will be:

```
MHB Labs - Private Access - (Preview)
17) Show Configurations and Status Private Access.
18) Configure Private Access.
19) Configure CSC Remote Management via Private Access.

e) Exit
Selection: 19 1

WARNING! You can isolate this node if the configuration is wrong.
Be careful with this settings. We recommend to be precise with the Host or Subnet configured here.
Subnet Prefixes less than /17 are not accepted.

No Management Networks are configured.

Do you want to configure Management Networks?

1) Yes
2) No
3) Reset to Default
Enter your choice: 1 2

Input Management Network (IP/Subnet Prefix): 172.25.0.0/24 3

Do you want to add another Management Network?

1) Yes
2) No
Enter your choice: 2 4

Management Networks to configure:

Management Networks Qty = 1
Management Network= 172.25.0.0/24

Do you want to apply changes?

1) Yes
2) No
Enter your choice: 1 5

Private Access - Management Network 172.25.0.0/24 was added on ns-cgc00004-a

Press Enter to continue...
```

You can add Multiple Networks

11 Remote Management using AWS and Rundeck

You can use several tools to Remote Manage the CSC. In this chapter, we are showing how to use AWS Systems Manager (Fleet Manager) and Rundeck.

11.1 AWS Systems Manager

The easiest and cheapest way to manage the Cloud Security Connectors is to use AWS Systems Manager. AWS official documentation is available here: <https://aws.amazon.com/systems-manager/>.

With AWS Systems Manager, you can manage the CSC remotely. To do it, you need to create "Documents" in advance. "Documents" are a series of commands used by the "Run Command" functionality.

This section explains how to create the "Documents" and how to "Run Commands".

11.1.1 Create Documents

We provide a CloudFormation template to create all "Documents" in one shot.

Steps:

1. Download the CloudFormation template from:

<https://mhb-netskope-cloudformation.s3.eu-west-1.amazonaws.com/MHB-Netskope-CSC-AWS-Systems-Manager-Documents-v-1-1.json>

The screenshot shows the AWS CloudFormation console's 'Create stack' wizard. The 'Specify template' step is selected. Red boxes and numbers 1-4 highlight the following elements:

- 1. The 'Create stack' button at the top.
- 2. The 'Template is ready' radio button under the 'Prepare template' section.
- 3. The 'Amazon S3 URL' radio button under the 'Specify template' section.
- 4. The 'Amazon S3 URL' text input field containing the URL: `https://mhb-netskope-cloudformation.s3.eu-west-1.amazonaws.com/MHB-Netskope-CSC-AWS-Systems-Manager-Documents-v-1-0.json`.

2. Deploy Stack. Go to Cloudformation → Create Stack
3. Insert the Amazon S3 URL and click next.
4. Put the Stack Name

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Specify stack details

Stack name

Stack name

Documents-CSC-Netskope

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

Cancel Previous **Next**

5. Click Next -> Next -> Create Stack.

6. Wait the Stack to complete.

Stacks (1)

Filter by stack name

Active View nested

Documents-CSC-Netskope

2021-10-14 18:26:24 UTC+0100

CREATE_COMPLETE

7. Now go to Services -> Systems Manager -> and click "Documents" and choose "Owned by me"

AWS Systems Manager > Documents

Owned by Amazon Owned by me Shared with me All documents

Documents

Search by keyword or filter by tag or attributes

Copy-AWS-RunShellScript Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-Refresh-Proxy-Bypass-URL Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-Refresh-Routed-Bypass-URL Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-Reload-Config-json Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1
MHB-CSC-Reload-High-Availability Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-Reload-Routed-Bypass-json Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-ShowConfigurationAndStatus Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-ShowLogCurrentMonth Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1
MHB-CSC-ShowLogLastSixMonths Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-SpeedTest Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-SwitchTunnels Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1	MHB-CSC-TraceRouteAndLatencyTest Document type: Command Owner: 544690173127 Platform types: Linux Default version: 1

8. Done!

11.1.2 Run Commands

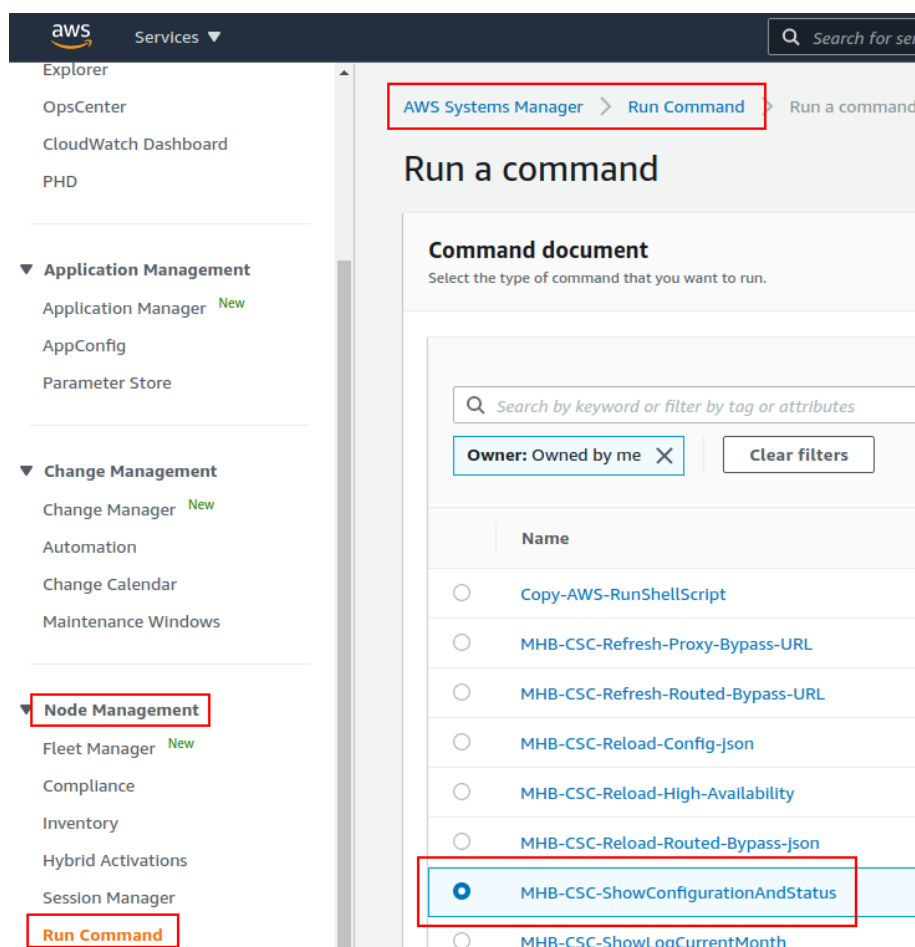
After you have created the Documents, you are ready to Run Commands on the CSC.

You can see the operation results on the "Output" section or store the results on S3 Buckets for further inspection.

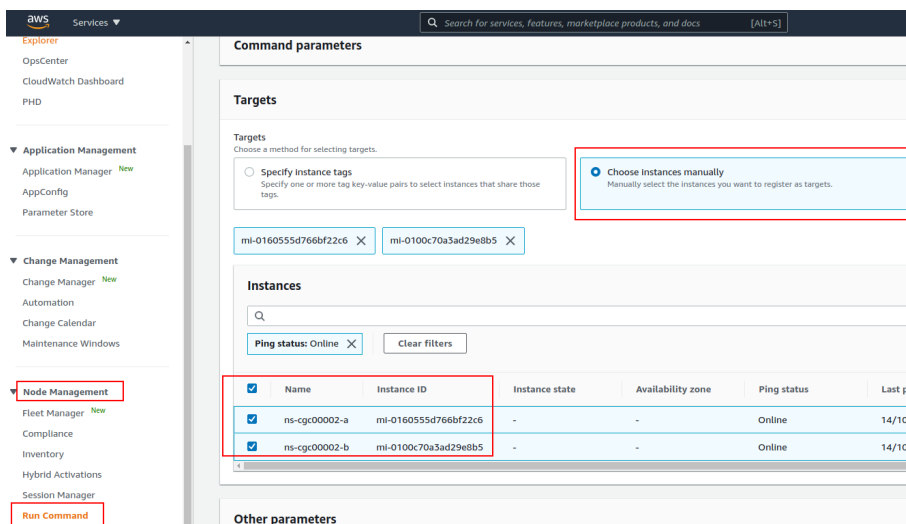
To "Run Commands", go to AWS Systems Manager → Instances & Nodes → Run Command.

Here is an example of Running: MHB-CSC-ShowConfigurationAndStatus

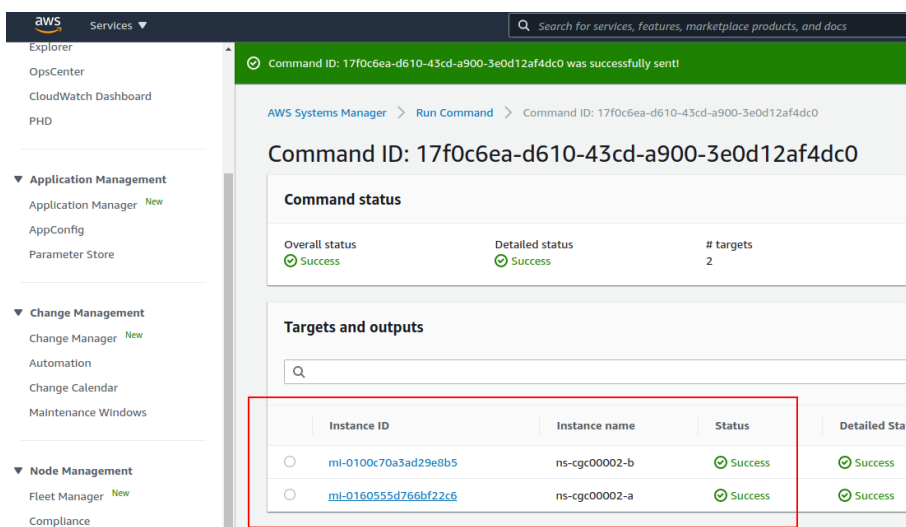
1. Run a Command
2. Select the Document created (Tip: Select "Owned by me")



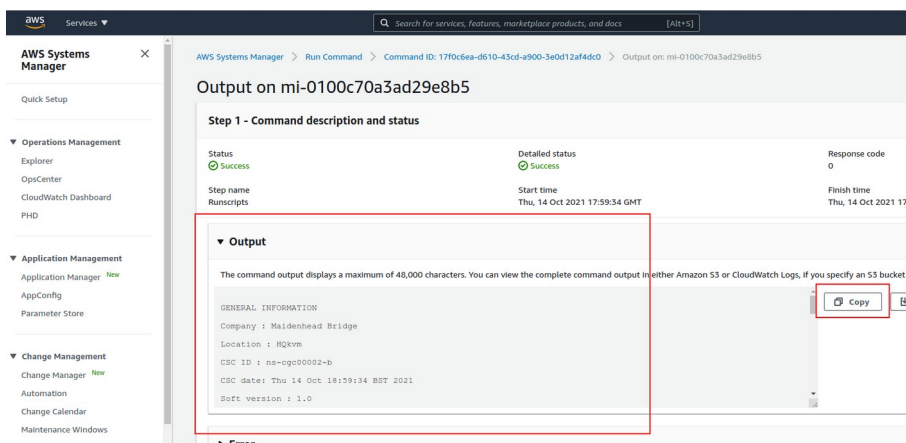
3. Scroll down and Select the Instances



4. Click "Run" . Wait for the Command Status "success"



5. Right click on Instance ID (mi-xxxx) and open in new tab. Check Output.



6. Done! (Note: You can copy the output and to display on a text editor for more visibility)

```
*Unsaved Document 1 x
1
2 GENERAL INFORMATION
3 Company : Maidenhead Bridge
4 Location : HQkvm
5 CSC ID : ns-cgc00002-b
6 CSC date: Thu 14 Oct 18:59:34 BST 2021
7 Soft version : 1.0
8
9 INTERFACES INFORMATION
10 External: Tunnel IP: 192.168.1.60 | Bypass Proxy Egress IP: 192.168.1.61 | CSC IP(eth0): 192.168.1.63/24 | Network Gateway: 192.168.1.240 is Alive
11 Internal: CSC GW IP: 172.19.0.60 | CSC IP(eth1): 172.19.0.64/24 | Network Gateway: 172.19.0.133 is Alive
12
13 TRAFFIC REDIRECTION Options
14 To Netskope: VIP Proxy: 172.19.0.61:80 | Route all traffic via CSC GW IP | Netskope Global Proxy IP: 163.116.128.80:80 via CSC GW IP
15 Direct to Internet: Bypass Proxy: 172.19.0.62:3128 | Netskope Global Proxy IP: 163.116.128.80:3128 via CSC GW IP
16
17 DNS INFORMATION
18 DNS Server (1) IP: 172.19.0.100 is Alive
19 DNS Server (2) IP: 1.1.1.1 is Alive
20
21 NETSKOPE INFORMATION
22 GRE tunnels egress Public IP: 82.68.6.74
23
24 Primary Tunnel:
25     Node : GB,London,LON1
26     Node Public IP: 163.116.162.36
27     Node Probe: 10.162.6.209
28 Secondary Tunnel:
29     Node : GB,Manchester,MAN1
30     Node Public IP: 163.116.165.36
31     Node Probe: 10.165.6.209
32
33 TUNNEL STATUS
34 Primary Tunnel (reachability):
35     Node Keepalive is: Alive
36     GRE Tunnel IP is: Standby - This CSC (ns-cgc00002-b) is Cluster Standby
37 Secondary Tunnel (reachability):
38     Node Keepalive is: Alive
39     GRE Tunnel IP is: Standby - This CSC (ns-cgc00002-b) is Cluster Standby
40 returnToPrimaryTunnel: true
41
42 Tunnel Status: No active tunnel since: Tue 5 Oct 19:27:15 UTC 2021
43
44 HTTP://WWW.NETSKOPE.COM PAGE STATUS
45 No test performed - This CSC (ns-cgc00002-b) is Cluster Standby
46
47 PROXY BYPASS - EGRESS INTERFACE STATUS
48 No test performed - This CSC (ns-cgc00002-b) is Cluster Standby
49
50 ROUTED BYPASS
51 Using Routed Bypass URL: https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json
52 Routed Bypass URL https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile.json is reachable
53 Routed Bypass Rules configured via URL: 8
54
55 AWS SSM AGENT
56 AWS SSM Agent is active (running) since Tue 2021-10-05 20:27:11 BST; 1 weeks 1 days ago
57 Registration values: {"ManagedInstanceID":"mi-0100c70a3ad29e8b5","Region":"eu-west-2"}
58
59 SYSLOG INFORMATION
60 SYSLOG Server (1) IP: 172.19.0.199 is Alive
61 SYSLOG Server (2) IP is not configured
62 SYSLOG TCP Port: 514
63
64 HIGH AVAILABILITY Information
65 This CSC (ns-cgc00002-b) is Cluster STANDBY
```

11.1.3 List of Documents available for "Run Command"

1. "MHB-CSC-ShowConfigurationAndStatus": Executes "Show Configuration and Status"
2. "MHB-CSC-SpeedTest": Performs speedtest.net on the CSC.
3. "MHB-CSC-TraceRouteAndLatencyTest": Performs MyTraceRoute test against the Primary and Secondary ZEN. It also does a Reverse Test from the tunnel active to your Public IP if the tunnel is up.
4. "MHB-CSC-Refresh-Proxy-Bypass-URL": Refresh the Proxy Bypass list using the values of the Proxy Bypass PAC file stored in the URL configured.
5. "MHB-CSC-Refresh-Routed-Bypass-URL": Refresh the Routed Bypass list using the values of the JSON file stored in the URL configured.
6. "MHB-CSC-ShowLogCurrentMonth": Shows current month logs.
7. "MHB-CSC-ShowLogLastSixMonths": Shows last six month logs.
8. "MHB-CSC-SwitchTunnels": Switch tunnels.
9. "MHB-CSC-Reload-Config-json": Reloads the values of config.json file.
10. "MHB-CSC-Reload-High-Availability": Reloads the values of highAvailability.json file. (for CSC on AWS, Azure and Gcloud. Not in use on CSC for Virtual Platforms.
11. "MHB-CSC-Reload-Routed-Bypass-json": Reloads the values of routedBypassRulesFile.json.
12. "MHB-CSC-Update-Nodes-Database": Updates the Netskope Node Database.
13. "MHB - CSC - Refresh Private Access Peers URL": Refresh the Private Access Peers list using the values of the JSON file stored in the URL configured.
14. "MHB - CSC - Reload Private Access Peers JSON file": Reloads the values of privateAccessPeersConfig.json
15. "MHB - CSC - Show Private Access ALL Peers status": Show the Status of all Private Access Peers.

11.2 Rundeck

Rundeck (<https://www.rundeck.com/>) is an open-source software Job scheduler and Run Book Automation system for automating routine processes across development and production environments. It combines task scheduling multi-node command execution workflow orchestration and logs everything that happens.

Installation Steps:

1. Install Rundeck. Instructions at: <https://www.rundeck.com/open-source>
2. Create a Project.
3. Enable user "csccli" and setup the SSH Public key on each CSC.
4. On the Project, setup the SSH Private and define the nodes:

The screenshot shows the Rundeck web interface. At the top, a dropdown menu is set to 'NS-CSC-MGMT' and the word 'Project' is displayed. Below this, the 'Edit Nodes File' section is active, showing the file path '/home/rundeck06/rundeck/NS-CSC-MGMT-NODES.json'. The 'Source' is '2. File Reads a file containing node definitions in a supported format', the 'Format' is 'json', and the 'Description' is '/home/rundeck06/rundeck/NS-CSC-MGMT-NODES.json'. A 'Soft Wrap' toggle is visible. The main area displays a JSON configuration for nodes. A red box highlights the first node definition, and a red '3' is next to it. The JSON defines four nodes: 'ns-cgc00002-a' (active), 'ns-cgc00002-b' (active), 'ns-cgc00001-a' (inactive), and 'ns-cgc00001-b' (inactive). Each node has fields for hostname, nodename, description, tags, username, osVersion, and osName. At the bottom left, a 'PROJECT SETTINGS' icon is highlighted with a red box. 'Cancel' and 'Save' buttons are at the bottom.

```
1 {
2   "ns-cgc00002-a": {
3     "hostname": "172.19.0.63",
4     "nodename": "ns-cgc00002-a",
5     "description": "CSC GRE Cluster A",
6     "tags": "csc-gre-cluster,netskope,active",
7     "username": "csccli",
8     "osVersion": "1.0",
9     "osName": "csc-gre-cluster"
10  },
11  "ns-cgc00002-b": {
12    "hostname": "172.19.0.64",
13    "nodename": "ns-cgc00002-b",
14    "description": "CSC GRE Cluster B",
15    "tags": "csc-gre-cluster,netskope,active",
16    "username": "csccli",
17    "osVersion": "1.0",
18    "osName": "csc-gre-cluster"
19  },
20  "ns-cgc00001-a": {
21    "hostname": "172.19.0.23",
22    "nodename": "ns-cgc00001-a",
23    "description": "CSC GRE Cluster A",
24    "tags": "csc-gre-cluster,netskope,inactive",
25    "username": "csccli",
26    "osVersion": "1.0",
27    "osName": "csc-gre-cluster"
28  },
29  "ns-cgc00001-b": {
30    "hostname": "172.19.0.24",
31    "nodename": "ns-cgc00001-b",
32    "description": "CSC GRE Cluster B",
33    "tags": "csc-gre-cluster,netskope,inactive",
34    "username": "csccli",
35    "osVersion": "1.0",
36    "osName": "csc-gre-cluster"
37  }
38 }
39 }
```

5. Create the jobs. Please, contact Support at <http://support.maidenheadbridge.com> for the latest Job List.

11.2.1 Jobs

The following screen shows the list of Jobs available.

NS-CSC-MGMT

17 All Jobs

Expand All Collapse All

- ▶ Check CSC Status - Netskope This test checks L7 Keepalives on CSCs using Netskope Cloud 🟢 in 11m
- ▶ Refresh Proxy Bypass URL
- ▶ Refresh Proxy Bypass URL - CSCs with tags:active This job executes Refresh Proxy Bypass List command on all CSCs with tags:active
- ▶ Refresh Routed Bypass URL This job updates the Routed Bypass Configuration on the CSC using the Routed Bypass URL.
- ▶ Refresh Routed Bypass URL - CSCs with tags:active This job updates the Routed Bypass Configuration on the CSCs with tags:active using the Routed Bypass URL
- ▶ Reload Config Json File This job reloads the values of the config.json file onto the CSC.
- ▶ Reload High Availability Json File This job is valid only for CSCs on AWS, Azure and Gcloud.
- ▶ Reload Routed Bypass Json File
- ▶ Show Configuration and Status This job provides all configuration and statuses information of the CSC.
- ▶ Show Configuration and Status - CSC with tags:active This job executes Show Configuration and Status command on all CSCs with tag:active
- ▶ Show Logs Current Month
- ▶ Show Logs Last 6 Months
- ▶ Speed Test This job executes Speed Test from the CSC to speedtest.net
- ▶ Switch Tunnels This Job Switches tunnels Primary / Secondary
- ▶ Test Email Use this job to check that you are receiving alerts via email.
- ▶ Traceroute and Latency Test Use this Job to check the quality of the path to the Cloud - hop by hop
- ▶ Update Nodes Database

11.2.2 Running job "Show Configuration and Status"

NS-CSC-MGMT

✓ Show Configuration and Status - CSC with tags:active ⌵ Succeeded 0:00:09 at 7:38 pm 👤 you

This job executes Show Configuration and Status command on all CSCs with tag:active

Log Output +

Node	Start time	Duration
ns-cgc00002-a		0:00:05
Command	7:38:08 pm	0:00:05

18:38:11 GENERAL INFORMATION

18:38:11 Company : Maidenhead Bridge

18:38:11 Location : HQkvm

18:38:11 CSC ID : ns-cgc00002-a

18:38:11 CSC date: Thu 14 Oct 19:38:10 BST 2021

18:38:11 Soft version : 1.0

18:38:11 INTERFACES INFORMATION

18:38:11 External: Tunnel IP: 192.168.1.60 | Bypass Proxy Egress IP: 192.168.1.61 | CSC IP(eth0): 192.168.1.62/24 | Network Gateway: 192.168.1.240 is Alive

18:38:11 Internal: CSC GW IP: 172.19.0.60 | CSC IP(eth1): 172.19.0.63/24 | Network Gateway: 172.19.0.133 is Alive

18:38:11 TRAFFIC REDIRECTION Options

18:38:11 To Netskope: VIP Proxy: 172.19.0.61:80 | Route all traffic via CSC GW IP | Netskope Global Proxy IP: 163.116.128.80:80 via CSC GW IP

18:38:11 Direct to Internet: Bypass Proxy: 172.19.0.62:3128 | Netskope Global Proxy IP: 163.116.128.80:3128 via CSC GW IP

18:38:11 DNS INFORMATION

18:38:11 DNS Server (1) IP: 172.19.0.100 is Alive

18:38:11 DNS Server (2) IP: 1.1.1.1 is Alive

18:38:11 NETSKOPE INFORMATION

18:38:11 GRE tunnels egress Public IP: 82.68.6.74

18:38:11 Primary Tunnel:

18:38:11 Node : GB,London,LON1

18:38:11 Node Public IP: 163.116.162.36

18:38:11 Node Probe: 10.162.6.209

18:38:11 Secondary Tunnel:

12 DevOps operations

The CSC is delivered with all configurations and is ready for production. Even so, during the life cycle of the CSC, some parametrization may be required to be changed or modified. For this reason, we provide some configuration utilities that will help with further parametrization and change management.

The CSC offers an option to do some changes using JSON config files. The operation is simple and is three steps:

1. Obtain the current JSON file from the CSC.
2. Download the modified JSON file to the CSC.
3. "Run Command" (AWS Systems Manager) of the specific "reload" document. (or use Rundeck Job)

The JSON files available are:

1. **config.json**: Allows administrators to modify specific values on the CSC like GRE Primary, Secondary nodes, DNS, Syslog, Routed Bypass URL, Proxy Bypass URL, Etc.
2. **routedBypassRulesFile.json**: Allows administrators to manually configure Routed Bypass Rules if not using the Routed Bypass URL method.
3. **privateAccessPeersConfig.json**: Use this Json file to configure "networks" and "privateApps" on your Private Cloud.

In this chapter, we are going to explain the procedures.

12.1 config.json file

You can use this file to change DNS, Log Servers, GRE Nodes (Primary/Secondary), etc.

1. Obtain the current "config.json" from the CSC, running "Run Command" (AWS-RunShellScript.). For example:

*The fields in **bold** are not configurable. So please, do not modify.*

```
cat /usr/local/etc/mhb-csc/config.json
```

```
{
  "model": "csc-gre-ns-vm",
  "version": "1.1",
  "dns": {
    "primaryDnsIP": "172.19.0.100",
    "secondaryDnsIP": "172.19.0.101"
  },
  "bypassProxyPublicIP": "82.68.6.73",
  "bypassProxyPacUrl": "https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/mhb-netskope-bypass-vm.pac",
  "syslogServers": {
    "primarySyslogIP": "172.19.0.199",
    "secondarySyslogIP": "",
    "syslogTcpPort": 514
  },
  "greCredentials": {
    "autoDiscovery": true,
    "grePublicIP": "82.68.6.74",
    "primaryGreGateway": "163.116.162.36",
    "primaryProbelpAddress": "10.162.6.209",
    "primaryLocation": "GB,London,LON1",
    "secondaryGreGateway": "163.116.165.36",
    "secondaryProbelpAddress": "10.165.6.209",
    "secondaryLocation": "GB,Manchester,MAN1"
  },
  "tunnelRedundancy": {
    "returnToPrimaryTunnel": true
  },
  "routedBypassPublicIP": "82.68.6.73",
  "routedBypassJsonFileUrl": "https://mhb-netskope-pac-files.s3.eu-west-1.amazonaws.com/routedBypassRulesFile-documentation.json",
  "privateAccesPublicIpPort": "82.68.6.74:51821",
  "privateAccesPeersJsonFileUrl": "https://mhb-netskope-private.s3.eu-west-1.amazonaws.com/privateAccessPeersConfig-LAB2.json",
  "privateAccessRemoteManagement": [
    "172.19.0.0/24"
  ]
}
```

2. Create a AWS bucket and place the modified "config.json" file on it.
3. Download the file to the CSC. Run Command "AWS-RunShellScript"

```
wget <Your bucket file URL> -O /usr/local/etc/mhb-csc/config.json
```

4. Run Document "MHB-CSC-Reload-Config-json" to apply the changes.


12.2 routedBypassRulesFile.json

You can use this file to create Routed Bypass Rules manually instead of using the automatic method via Routed Bypass URL.

1. Obtain the current "routedBypassRulesFile.json" from the CSC, running "Run Command" (AWS-RunShellScript.). For example:

```
cat /usr/local/etc/mhb-csc/routedBypassRulesFile.json
```

```
{
  "routedBypassRules": [
    {
      "description": "O365 Login URLs 1",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "20.190.128.0/18",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "O365 Login URLs 2",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "20.190.128.0/18",
      "fromPort": "443",
      "toPort": "443"
    },
    {
      "description": "O365 Login URLs 3",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "40.126.0.0/18",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "portquiz.net",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.47.209.216/32",
      "fromPort": "80",
      "toPort": "80"
    },
    {
      "description": "O365 Login URLs 4",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "40.126.0.0/18",
      "fromPort": "443",
      "toPort": "443"
    },
    {
      "description": "Skype and Teams UDP 1",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "13.107.64.0/18",
      "fromPort": "3478",
      "toPort": "3481"
    },
    {
      "description": "Skype and Teams UDP 2",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.112.0.0/14",
      "fromPort": "3478",
      "toPort": "3481"
    },
    {
      "description": "Skype and Teams UDP 3",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "52.120.0.0/14",
      "fromPort": "3478",
      "toPort": "3481"
    }
  ]
}
```

- 
2. Create a AWS bucket and place on it the modified "routedBypassRulesFile.json" file.
 3. Download the file to the CSC. Run Command "AWS-RunShellScript"

```
wget <Your bucket file URL> -O /usr/local/etc/mhb-csc/routedBypassRulesFile.json
```

4. Run Document "MHB-CSC-Reload-Routed-Bypass-json" to apply the changes.

12.3 privateAccessPeersConfig.json

You can use this file to create Private Access Peer Rules manually instead of using the automatic method via Private Access Peers URL.

1. Obtain the current "privateAccessPeersConfig.json" from the CSC, running "Run Command" (AWS-RunShellScript.). For example:

```
cat /usr/local/etc/mhb-csc/privateAccessPeersConfig.json
```

```
{
  "peers": [
    {
      "nodeName": "ns-cgc00001",
      "description": "Node on VMware Server 1",
      "location": "HQ",
      "publicKey": "yAnz5TF+XXJte14tj3zlMNq+hd2rYUlgJBgB3fBmk=",
      "publicIpAndUdpPort": "200.1.1.1:51821",
      "privateCirdIp": "192.168.7.1/24",
      "persistentKeepAlive": "no",
      "networks": [ "10.1.1.0/24", "10.1.2.0/24" ],
      "privateApps": [
        {
          "description": "Allow all traffic to this site",
          "ipProtocol": "all",
          "sourceCirdIp": [ "0.0.0.0/0" ],
          "destinationCirdIp": [ "10.1.1.0/24", "10.1.2.0/24" ],
          "destinationSinglePorts": [ "" ],
          "destinationPortRange": { "fromPort": "", "toPort": "" }
        }
      ]
    },
    {
      "nodeName": "ns-cgc00002",
      "description": "Node on VMware Server 2",
      "location": "Datacentre 2",
      "publicKey": "xTlBASrboUvnH4htodjb6e697QjLErt1NAB4mZqp8Dg=",
      "publicIpAndUdpPort": "200.1.1.2:51821",
      "privateCirdIp": "192.168.7.2/24",
      "persistentKeepAlive": "no",
      "networks": [ "10.2.1.0/24", "10.2.2.0/24" ],
      "privateApps": [
        {
          "description": "Allow all traffic to this site",
          "ipProtocol": "all",
          "sourceCirdIp": [ "0.0.0.0/0" ],
          "destinationCirdIp": [ "10.2.1.0/24", "10.2.2.0/24" ],
          "destinationSinglePorts": [ "" ],
          "destinationPortRange": { "fromPort": "", "toPort": "" }
        }
      ]
    },
    {
      "nodeName": "ns-cgc00003",
      "description": "Node on VMware Server 3",
      "location": "Branch",
      "publicKey": "TrMvSoP4jYQlY6RlZBgbsQqY3vxl2Pi+y71IOWWXX0=",
      "publicIpAndUdpPort": "200.1.1.3:51821",
      "privateCirdIp": "192.168.7.3/24",
      "persistentKeepAlive": "no",
      "networks": [ "10.3.1.0/24", "10.3.2.0/24" ],
      "privateApps": [
        {
          "description": "Allow all traffic to this site",
          "ipProtocol": "all",
          "sourceCirdIp": [ "0.0.0.0/0" ],
          "destinationCirdIp": [ "10.3.1.0/24", "10.3.2.0/24" ],
          "destinationSinglePorts": [ "" ],
          "destinationPortRange": { "fromPort": "", "toPort": "" }
        }
      ]
    }
  ]
}
```

2. Create a AWS bucket and place on it the modified "privateAccessPeersConfig.json" file.

- 
3. Download the file to the CSC. Run Command "AWS-RunShellScript"

```
wget <Your bucket file URL> -O /usr/local/etc/mhb-csc/privateAccessPeersConfig.json
```

4. Run Document "MHB-CSC-Reload-Private-Access-JSON-file" to apply the changes.

13 Appendixes

13.1 Appendix A: Routed Bypass JSON file if you don't have Cloud Firewall License.

If you decide to configure the default route to the internet via the CSC and don't have a Cloud Firewall license, you need to send only HTTP and HTTPS via the GRE tunnel and the rest of the traffic via Routed Bypass.

The following JSON file does the work to redirect only Web traffic via the GRE tunnel, and the rest goes directly via the Bypass Interface.

```
{
  "routedBypassRules": [
    {
      "description": "Bypass ICMP all",
      "ipProtocol": "icmp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "0.0.0.0/0",
      "fromPort": "",
      "toPort": ""
    },
    {
      "description": "Bypass TCP Ports I",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "0.0.0.0/0",
      "fromPort": "1",
      "toPort": "79"
    },
    {
      "description": "Bypass TCP Ports II",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "0.0.0.0/0",
      "fromPort": "81",
      "toPort": "442"
    },
    {
      "description": "Bypass TCP Ports III",
      "ipProtocol": "tcp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "0.0.0.0/0",
      "fromPort": "444",
      "toPort": "65535"
    },
    {
      "description": "Bypass UDP Ports all",
      "ipProtocol": "udp",
      "sourceCirdIp": "0.0.0.0/0",
      "destinationCirdIp": "0.0.0.0/0",
      "fromPort": "1",
      "toPort": "65535"
    }
  ]
}
```



13.2 Appendix B: Release Notes

13.2.1 Version 1.0

This is the initial version of the Cloud Security Connector for Netskope.

13.2.2 Version 1.1

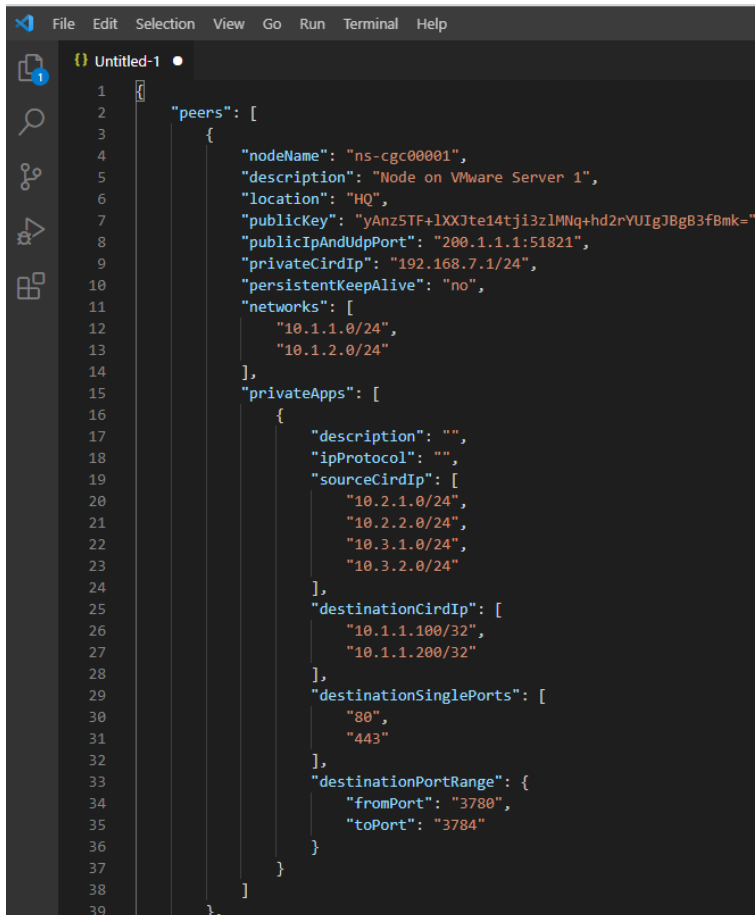
The version 1.1 contains the following enhancements:

- New! Private Cloud Private Access (PriCPA) functionality. PriCPA allows you to create a Private Cloud among all CSCs for private traffic. In a matter of minutes, you can build a full mesh between your locations for private traffic with Zero Trust. After making the Private Cloud, you can set up your policies to define who will talk with who inside your Private Cloud.
- Minor cosmetic changes on some menus.

13.3 Appendix C: JSON formatters (Visual Code, Notepad ++)

We strongly recommend using Software that can show errors on your JSON file and also can format (beautify) the file for better visibility. Below two examples.

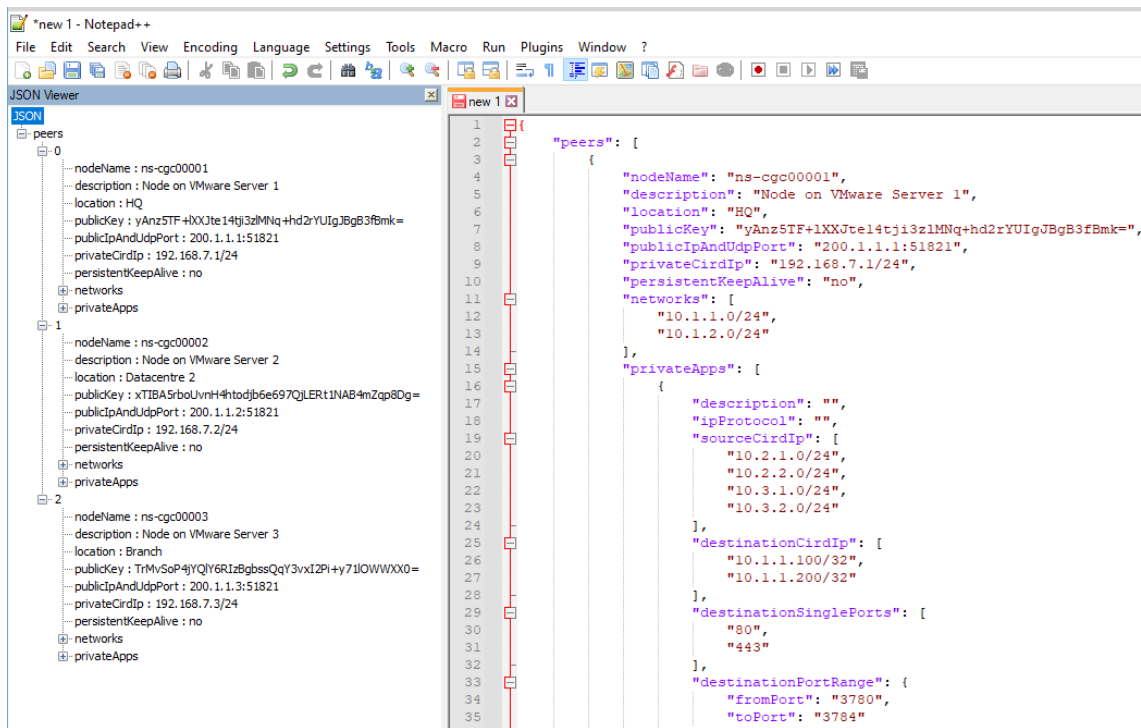
13.3.1 Visual Code



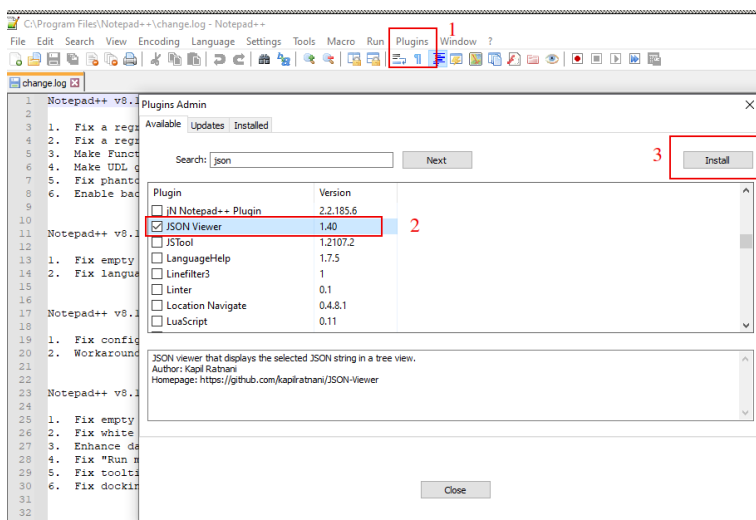
```
1  {
2    "peers": [
3      {
4        "nodeName": "ns-cgc00001",
5        "description": "Node on VMware Server 1",
6        "location": "HQ",
7        "publicKey": "yAnz5TF+lXXJte14tji3zIMNq+hd2rYUigJBg83fBmk=",
8        "publicIpAndUdpPort": "200.1.1.1:51821",
9        "privateCirdIp": "192.168.7.1/24",
10       "persistentKeepAlive": "no",
11       "networks": [
12         "10.1.1.0/24",
13         "10.1.2.0/24"
14       ],
15       "privateApps": [
16         {
17           "description": "",
18           "ipProtocol": "",
19           "sourceCirdIp": [
20             "10.2.1.0/24",
21             "10.2.2.0/24",
22             "10.3.1.0/24",
23             "10.3.2.0/24"
24           ],
25           "destinationCirdIp": [
26             "10.1.1.100/32",
27             "10.1.1.200/32"
28           ],
29           "destinationSinglePorts": [
30             "80",
31             "443"
32           ],
33           "destinationPortRange": {
34             "fromPort": "3780",
35             "toPort": "3784"
36           }
37         }
38       ]
39     },
40   ]
41 }
```

1. Download : <https://code.visualstudio.com/download>
2. Select your platform and install.
3. Create your JSON.
 - 3.1. Visual Code will show the errors in RED.
 - 3.2. To "Beautify" your JSON file press:
 - 3.2.1. On Windows: "Shift + Alt + F"
 - 3.2.2. On MAC: "Shift + Option + F"
 - 3.2.3. On Linux: " Ctrl + Shift + I"

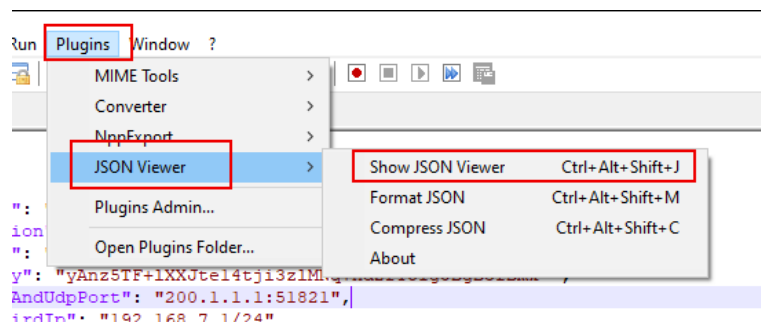
13.3.2 Notepad ++



1. Download: <https://notepad-plus-plus.org/downloads/>
2. Install JSON Viewer Plug in.



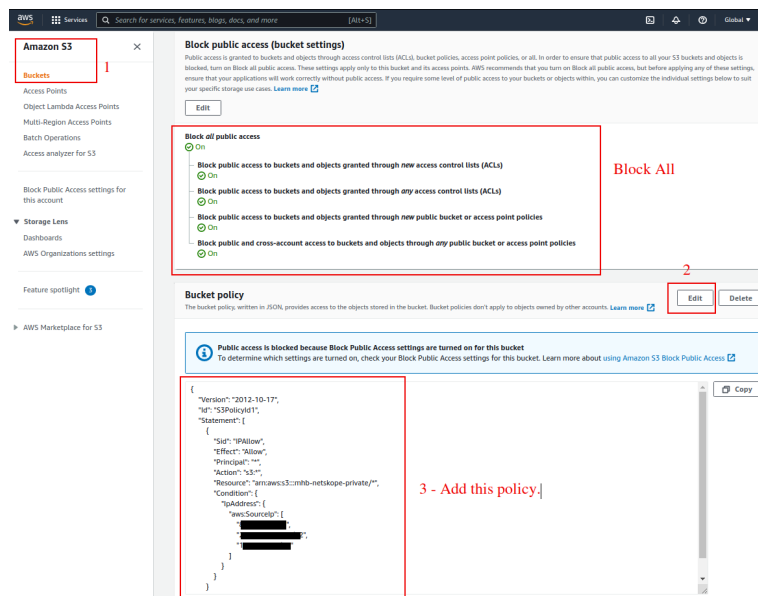
3. Create your JSON file.
4. To Check your JSON file go to: Plugins -> JSON Viewer -> Show JSON Viewer.



5. To format ("Beautify") your JSON go to: Plugins -> JSON Viewer -> Format JSON

13.4 Appendix D: Securing an AWS Bucket by source IP.

1. On your AWS console create a bucket with default values for permissions: "Block a// Public Access = on"
2. On Bucket Policy, add your Public IPs in "aws:SourceIp":[]



```
{
  "Version": "2012-10-17",
  "Id": "S3PolicyId1",
  "Statement": [
    {
      "Sid": "IPAllow",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::mhb-netskope-private/*",
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "200.1.1.1/32",
            "200.1.1.2/32",
            "200.2.0.0/24"
          ]
        }
      }
    }
  ]
}
```

3. Done!