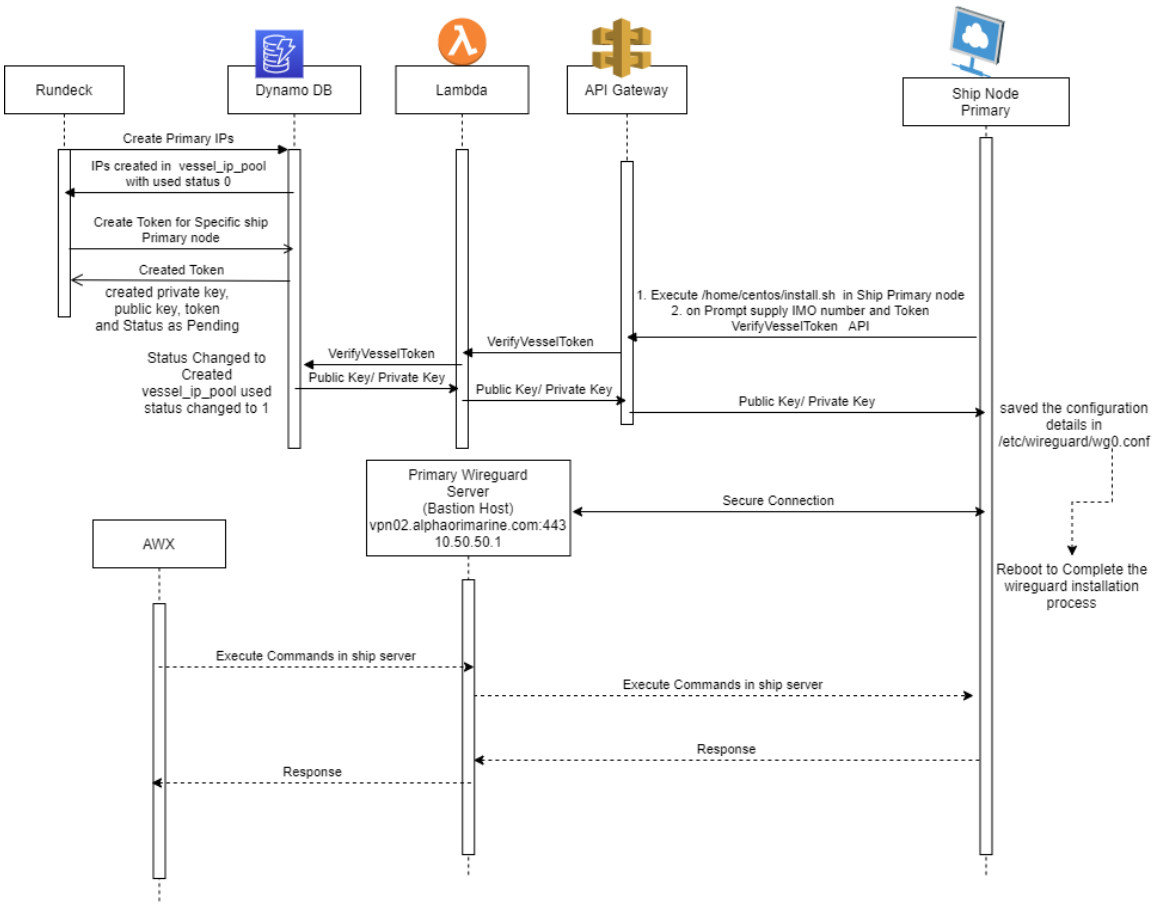
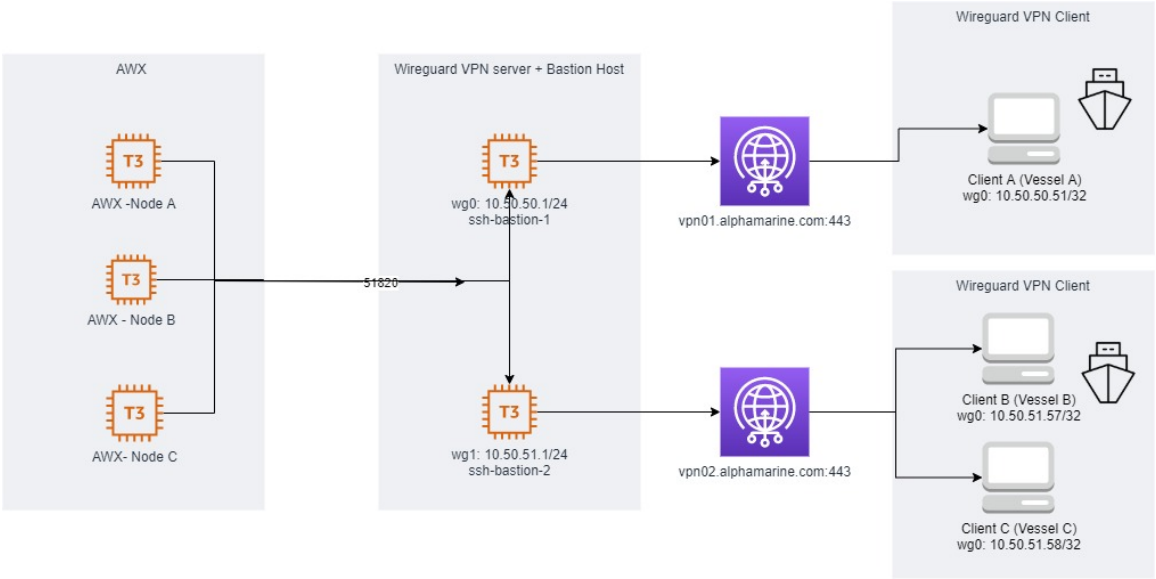
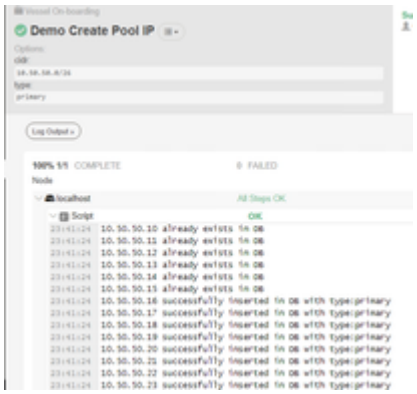
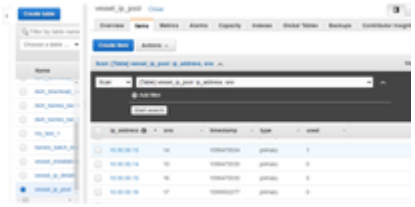
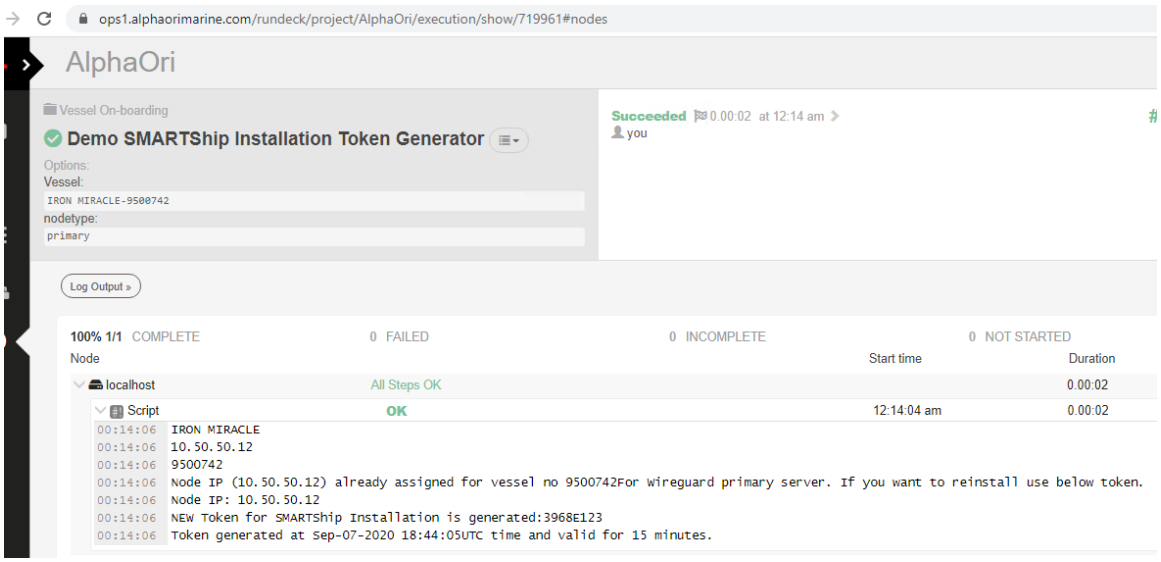
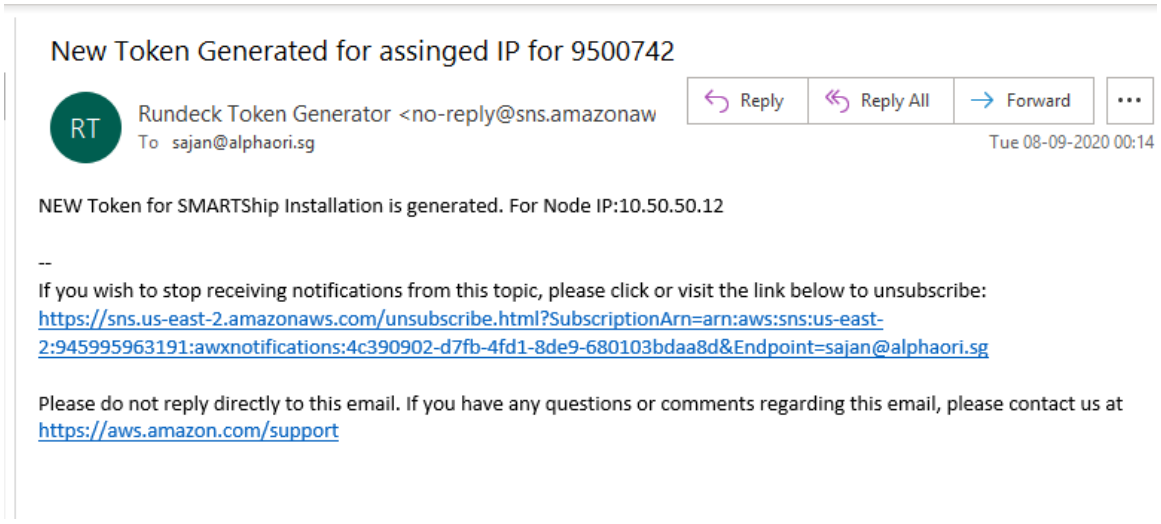
Wireguard Testing

Resources	Path
Dynamo DB vessel Ip details	https://us-east-2.console.aws.amazon.com/dynamodb/home?region=us-east-2#tables:selected=vessel_ip_details;tab=items
Ansible AWX	https://awx.alphaorimarine.com/#/jobs?job_search=page_size:20;order_by:-finished;not__launch_type:sync
wireguard test cases	https://docs.google.com/spreadsheets/d/1qwaVes6leSyR6Zl8pitszqn9XYpvlhE9xxrTxAtuqAc/edit?ts=5f55bd63#gid=0
Rundeck job for installation Token Generation	https://ops1.alphaorimarine.com/rundeck/project/AlphaOri/job/show/15ae7128-4399-47a3-885f-cbab118d2b3a
Wireguard architechture detailed	

Wireguard architecture for SMARTShip	
Creating IP Pool and to verify in the Dynamo DB	  As per the IP address count, `10.50.50.0/26`, 64 address location should be created. But actually, it was only 54(6 already created + 48 created new + 10 reserved IP addresses)

Token Generation, if the vessel already exists





Token Generation, for new vessel

ops1.alphaorimarine.com/rundeck/project/AlphaOri/execution/show/719964#nodes

Vessel On-boarding

Demo SMARTShip Installation Token Generator

Succeeded 0.00:

you

Options:

Vessel:

FURANO GALAXY-9804928

nodetype:

primary

Log Output »

100% 1/1 COMPLETE 0 FAILED 0 INCOMPL

Node

localhost All Steps OK

Script OK

```
00:16:33 FURANO GALAXY
00:16:33 10.50.50.14
00:16:33
00:16:33 9804928
00:16:33 Token for SMARTShip Installation is 4A555426
00:16:33 Token generated at Sep-07-2020 18:46:32UTC time and valid for 15 minutes.
00:16:33 Updated Client 10.50.50.14 IP on wireguard Primary server
```

Scan: [Table] vessel_ip_details: ref_key

Viewing 1 to 5 items

Scan [Table] vessel_ip_details: ref_key

Add filter

Start search

ref_key	details	timestampcheck	vessel_no
1599475617	{ "nodeip" : { "S" : "10.50.50.10" }, "nodename" : { "S" : "BW HAZEL" }, "nod...	1599475617	9626687
1599476484	{ "nodeip" : { "S" : "10.50.50.11" }, "nodename" : { "S" : "BW CANOLA" }, "n...	1599479220	9687124
1599479982	{ "nodeip" : { "S" : "10.50.50.12" }, "nodename" : { "S" : "IRON MIRACLE" },...	1599504245	9500742
1599480800	{ "nodeip" : { "S" : "10.50.50.13" }, "nodename" : { "S" : "BW YUSHI" }, "nod...	1599480800	9810044
1599504392	{ "nodeip" : { "S" : "10.50.50.14" }, "nodename" : { "S" : "FURANO GALAXY...	1599504392	9804928

New Token generated for FURANO GALAXY

RT Rundeck Token Generator <no-reply@sns.amazonaw

To sajan@alphaori.sg

Reply Reply All Forward

Tue 08-09-2020 00

Check Rundeck job for new Token generated at Sep-07-2020 18:46:32UTC time and valid for 15 minutes.

--

| Useful commands | ```
systemctl restart wg-quick@tun6.service
sudo systemctl status wg-quick@wg0    # to check whether the wireguard is running
vi /etc/wireguard/wg0.conf            # path for the wireguard config file
sh /home/centos/install.sh            #to setup the wireguard configuration for a particular vessel
```<br>systemctl restart wg-quick@tun6.service |
| install.sh script | ```
  read -p "Enter Vessel IMO Number: " IMO
  read -p "Enter Installation Token: " TOKEN
  #MAC = `cat /sys/class/net/eth0/address`
  KEY=$(curl -sb -H "Accept: application/json" "https://1d1pq5w40k.execute-api.ap-south-1.amazonaws.com/VerifyVesselToken?vessel=$IMO&token=$TOKEN&mac=$MAC"| grep -i "ary")

  if [[ "$KEY" == "false" ]]; then
      echo "Invalid IMO Number or Token! Installation Failed!!"
      exit
  fi


  IFS='|' read -r -a ADDR <<< "$KEY"
  NODE_IP=${ADDR[0]}
  NODE_NAME=${ADDR[1]}
  NODE_TYPE=${ADDR[2]}
  CLIENT_KEY=${ADDR[3]}
  CLIENT_PUB_KEY=${ADDR[4]}
  WG_PUB_KEY=${ADDR[5]}

  sudo yum install epel-release https://www.elrepo.org/elrepo-release-7.el7.elrepo.noarch.rpm -y
  #sudo yum install yum-plugin-elrepo
  sudo yum install wireguard-dkms wireguard-tools kmod-wireguard -y

  #sudo sh -c 'wg genkey | tee privatekey | wg pubkey > publickey'
  sudo mkdir /etc/wireguard
  if [[ "$NODE_TYPE" == "primary" ]];
  then
  sudo -- bash -c '/bin/cat <<EOF >/etc/wireguard/tun6.conf
  [Interface]
  Address = wg_client_ip/32
```| |

```
  PrivateKey = client_private_key

  [Peer]
  PublicKey = server_public_key
  AllowedIPs = 10.50.40.1/24
  Endpoint = stagevpn01.alphaorimarine.com:443
  PersistentKeepalive = 15

  EOF'
  fi
  if [[ "$NODE_TYPE" == "secondary" ]];
  then
  sudo -- bash -c '/bin/cat <<EOF >/etc/wireguard/tun6.conf
  [Interface]
  Address = wg_client_ip/32
  PrivateKey = client_private_key

  [Peer]
  PublicKey = server_public_key
  AllowedIPs = 10.50.41.1/24
  Endpoint = stagevpn02.alphaorimarine.com:443
  PersistentKeepalive = 15

  EOF'
  fi

  sudo sed -i "s/wg_client_ip/$NODE_IP/" /etc/wireguard/tun6.
conf
  sudo sed -i "s%client_private_key%$CLIENT_KEY%" /etc
/wireguard/tun6.conf
  sudo sed -i "s%server_public_key%$WG_PUB_KEY%" /etc
/wireguard/tun6.conf


  sudo systemctl enable wg-quick@tun6.service

  read -p "Reboot machine now? " REBOOT_CONFIRM
  if [[ "$REBOOT_CONFIRM" != "Y" ]]; then
      echo "Reboot is required! You may please reboot
manually"
      exit
  fi
  sudo reboot
```

stage wireguard install.sh script	wg_client_install.sh

Footnotes

Term	Meaning/Resource
Global Accelerator	Improve **global** application availability AWS **Global Accelerator** continually monitors the health of your application endpoints, such as your Network Load Balancers, Application Load Balancers, EC2 Instances, or Elastic IPs, instantly reacting to changes in their health or configuration.
WireGuard	WireGuard is a free and open-source software application and communication protocol that implements virtual private network techniques to create secure point-to-point connections in routed or bridged configurations
Ansible Playbook	*Playbooks are Ansible's configuration, deployment, and orchestration language. They can describe a policy you want your remote systems to enforce, or a set of steps in a general IT process.* *If Ansible modules are the tools in your workshop, playbooks are your instruction manuals, and your inventory of hosts are your raw material.* **Playbooks** are the files where **Ansible** code is written. ... **Playbooks** are one of the core features of **Ansible** and tell **Ansible** w hat to execute. They are like a to-do list for **Ansible** that contains a list of tasks. **Playbooks** contain the steps which the user wants to execute on a particular machine.
bastion host	A **bastion host** is a special-purpose computer on a network specifically designed and configured to withstand attacks. The computer generally **hosts** a single application, for example a proxy server, and all other services are removed or limited to reduce the threat to the computer.
CIDR	https://cidr.xyz/ Classless_Inter-Domain_Routing Classless inter-domain routing (**CIDR**) is a set of Internet protocol (IP) standards that is used to create unique identifiers for networks and individual devices. The IP addresses allow particular information packets to be sent to specific computers.
Why do we need to create a pool of IPs in he beginning itself. why can't we create it in runtime when in need?	• On each peer, create a WireGuard interface and assign an IP address to it with the `ip` tool. It's an address inside a VPN network bound to the peer forever • On each peer generate a private key using the `wg` tool and assign it to the WireGuard interface • Derive a public key, again with the `wg` tool, and add it to all other peers you want to communicate. WireGuard doesn't specify how to exchange the keys. I opened an SSH session on each device and copied them over manually • Optionally, tell each peer how to reach other peers by specifying a public IP (or domain) and a port. Not all peers need to know how to reach others, as long as others know how to reach them; you'll see later

References

Resource	Path
Wireguard real world usage scenarios	https://www.zahradnik.io/wireguard-a-vpn-with-real-world-usage-in-mind
Wireguard use cases and commands	https://www.ivpn.net/knowledgebase/255/Linux---Autostart-WireGuard-in-systemd.html
wireguard comparison	https://www.ivpn.net/pptp-vs-ipsec-ikev2-vs-openvpn-vs-wireguard