Remote and Mobile Assets—Remote Site Management

Design and Implementation Guide for Managing Devices at Remote Sites

Published: August 2019

Contents

Remote and Mobile Assets—Remote Site Management

This module is part of the larger Remote and Mobile Assets (RaMA) Cisco Validated Design (CVD). Refer to the other modules for additional details about certain aspects of the architecture that are touched on in this module. All of the RaMA CVD modules are available at: www.cisco.com/go/rama

- **Solution Brief—An overview of the RaMA CVD and the available modules.**
- **Design and Implementation Guide (DIG)—Overall document for architecture, design, and best practice recommendations for remote and mobile asset deployments.**
- **Technology Guidance Module—Overview of the available hardware options for IoT gateways in the RaMA solution, with recommendations on hardware platform and software features to use for common scenarios.**
- **Security Module—Describes how the RaMA solution was designed from the ground up with security in mind. Includes detailed descriptions of how the solution fits into the SAFE model, including securing the gateways, data plane, and management plane. Also includes a section on achieving PCI compliance.**
- **Enterprise Network Integration Module—Best practices for the enterprise headend focusing on resiliency, high-availability, load-balancing, and security. Includes detailed descriptions of FlexVPN and WAN redundancy mechanisms.**
- **Fleet Management Module—Architecture for mobile applications in which the IR829 acts as the managed gateway and provides wired and wireless connectivity for southbound devices, as well as numerous northbound interfaces (LTE, Wireless Workgroup Bridge, GPS). Use of edge compute in the form of Cisco IOX is also included.**
- **Zero Touch Provisioning Module—Use of Kinetic GMM by IT personnel for provisioning and managing Cisco Industrial Routers with a focus on secure, scalable deployment.**
- **Field Deployment Module—Use of Kinetic GMM by OT personnel for deploying Cisco Industrial Routers in the field, with minimal knowledge of the underlying networking technology required.**
- **Edge Compute Module—Overview of the edge compute capabilities in Cisco Industrial Routers in the form of IOx. Includes implementation examples for deploying Dockerized applications.**

This module includes the following sections:

Overview

This module is targeted at remote asset owners and operators in industrial and commercial locations. The ability to collect data from machines and devices remotely can enable new business opportunities, improve efficiencies, and reduce cost. However this access to machines, remote kiosks, and remote SCADA devices must be managed to protect the security of the company's networks and to manage operational costs. This Remote Site CVD addresses these issues by leveraging the power of Cisco IOS-enabled routers with a rich set of features and an unparalleled emphasis on IT security. Additionally, the Cisco Kinetic Gateway Management Module (GMM) ensures simplified deployment of devices to thousands of locations using existing field personnel through zero touch deployment of gateways, template-based configurations, and several options for remote access using VPN.

Requirements

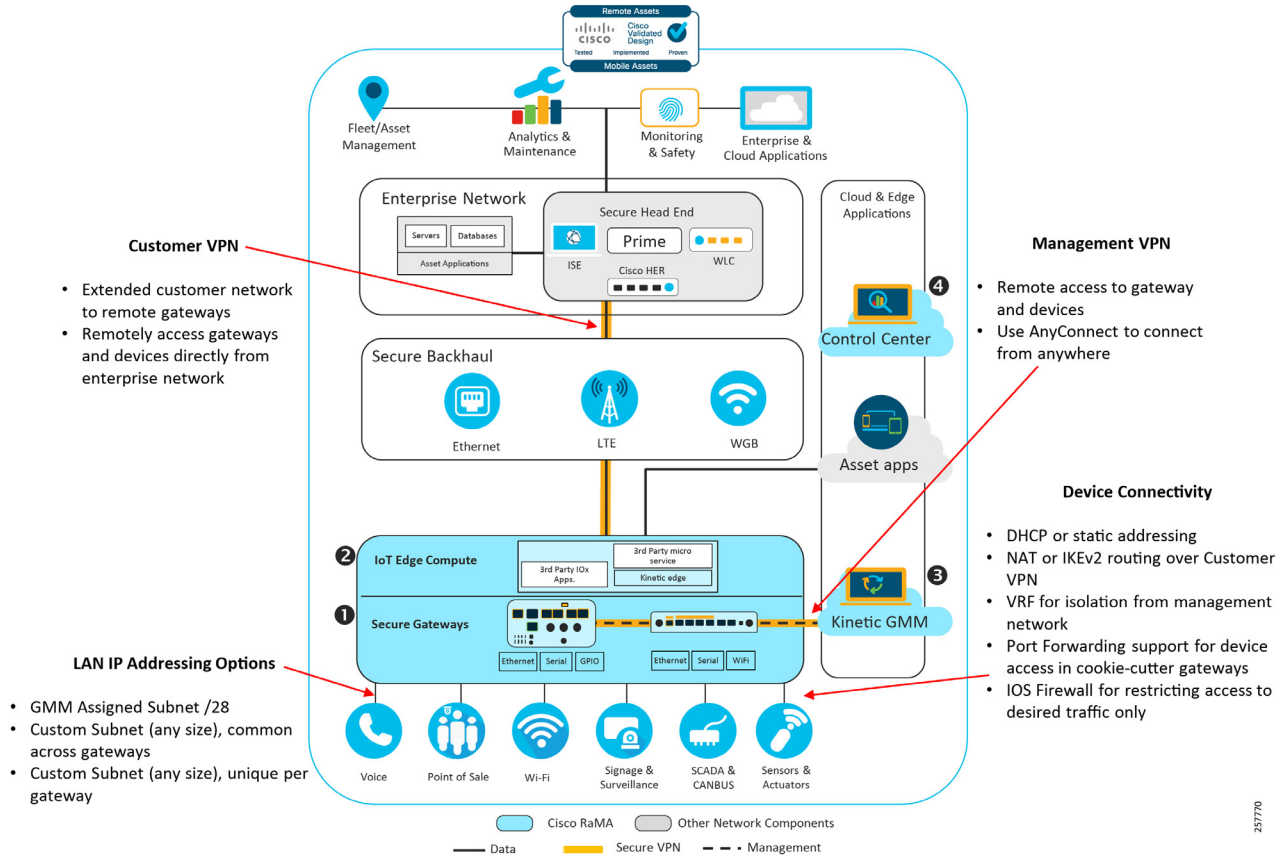
Requirements considered for this module include:

- Outbound connectivity for devices
- Inbound connectivity for devices behind the gateway
- Must work in conjunction with various WAN interfaces and providers
- Security, including isolation of management and data planes, and whitelisting of applications and devices
- Ability to easily and repeatably identify, address, access, install, and manage remote devices

Architecture

The RaMA architecture is the foundation for this module. Remote sites specifically require secure connectivity for the devices behind the gateway and often require connectivity to the gateway (inbound) as well. Securely providing options for these connectivity requirements using a managed gateway is the goal of this architecture.

Figure 1 RaMA Remote Site Architecture



Design

To understand the design options and recommendations, it is important to review some basic network concepts and product features and their use in the solution.

Multiple Interface Options

The managed gateways suitable for remote sites include a variety of interface options (dependent on the specific model), but generally include wired Ethernet (LAN and WAN), cellular (WAN), and WiFi (LAN and WAN). Additionally, serial ports and general-purpose input/output (GPIO) ports are available for connectivity to legacy or industrial devices.

Refer to the other RaMA solution modules for additional details on interface usage and best practices. For example, the Fleet module includes a thorough explanation of the available options for WiFi connectivity as both a hotspot (LAN) and WGB (WAN) interface.

Most of the concepts discussed in this document are agnostic to the specific interface type used for LAN or WAN connectivity.

IP Addressing for LAN devices

There are two primary options for configuring the LAN subnet for gateways in the RaMA solution:

- The subnet can either be assigned by Cisco Kinetic GMM itself and the devices will be reachable across the management VPN. This Cisco Kinetic GMM subnet will always have a subnet mask of /28, which means that 14 IP addresses will be available to end devices. For a small number of connected end devices, this is the simplest approach.
- If a larger number of end devices need to be connected to all or some of the gateways, a custom subnet solution can be defined, either across all gateways (templated in default settings) or uniquely assigned to each gateway (advanced setting). The advantage of using custom subnets is that they can be configured to use any size IP subnet mask, which translates into a larger supported end device population on the gateway. As an example, consider a remote site homing many sensors or other end devices that are connected via WiFI or an external switch behind the gateway. Using a custom subnet, this larger population of sensors and devices can be accommodated without the need for additional gateways.

Network Address Translation

Network Address Translation (NAT) is a common technique for dealing with several typical issues when connecting different networks, including topology hiding and the conservation of available IP addresses. NAT enables private IP networks that use nonregistered (RFC1918) “private” IP addresses to connect to the internet. The NAT function or IP address translation occurs in the gateway/router connecting two networks. Before packets are forwarded onto another network, NAT translates the private (not globally unique) IP addresses in the internal network into legal public IP addresses.

From a security standpoint, NAT can be configured to advertise to the outside world only one address for the entire network. This capability provides more security by effectively hiding the entire internal network behind that one address so that details of the internal network are hidden. Using the same (NATed) subnet across many sites also enables a single Cisco Kinetic GMM template to be used across all sites and for devices behind the gateway to use common IP addresses, which can aid in consistency, the quality of the installation, and ongoing access.

Port Forwarding

Expanding on the idea of NAT, port forwarding is a specific form of the technology in which the translation is extended to the Layer 4 (TCP or UDP) port number. This feature is commonly used to allow services running on specific TCP or UDP ports (such as TCP/80 for HTTP) on internal devices behind a gateway, with private IP addresses, to be made accessible from the outside. An outside user or application can connect to a specific port on the router’s public interface using a prescribed port number, which the router then translates (or forwards) onward to the destination device on its specific port number and IP address.

VPN—Management and Customer

As discussed in detail in the Enterprise Network Integration module, the RaMA solution provides a Management FlexVPN tunnel between Cisco Kinetic GMM and each managed gateway. Additionally, an optional FlexVPN tunnel can be configured between the managed gateway and the customer-owned VPN headend. Both tunnels provide secure, encrypted data transport for management and data plane traffic.

Routing with IKEv2

When utilizing the option to create a FlexVPN tunnel from the managed gateway to the enterprise VPN headend, the underlying Internet Key Exchange Protocol version 2 (IKEv2) protocol is able to advertise networks to the remote side (bidirectionally, from hub to spoke and spoke to hub). This functionality enables each spoke (managed gateway) to advertise unique networks to the enterprise via the headend, making the network behind the gateway directly reachable across the FlexVPN tunnel to the enterprise headend. This option can be somewhat more complicated to deploy, as each remote site (gateway) needs to be individually configured for its unique subnet.

However, this option gives enterprises granular control over which traffic from the remote gateways will traverse the enterprise headend network. If only some traffic needs to go to the enterprise, then a specific route can be advertised to the remote site, allowing all other traffic (the default route) to exit the gateway's WAN interface directly. In more security focused networks, the headend router can advertise a default route to the remote gateway, ensuring that all traffic destined for the enterprise headend (except for Cisco Kinetic GMM management traffic) traverses the FlexVPN tunnel. At the headend, comprehensive security applications and policies can be deployed to restrict, analyze, or monitor all remote site traffic.

Virtual Routing and Forwarding

Virtual Routing and Forwarding (VRF) is a feature that enables a Cisco router to be logically separated into multiple routing domains, each one isolated from the other. This is done by allowing multiple instances of a routing table to exist in a router, which work simultaneously and independently, increasing functionality by allowing network paths to be segmented (into domains) without the need for additional routers. This not only provides additional security, but permits the use of overlapping IP address space between the routing domains.

Advanced Templates

For the more experienced and skilled user, Cisco Kinetic GMM includes a feature that allows users with proper privileges to upload advanced templates and apply them to gateway templates. These advanced templates are essentially text-based IOS configuration lines, enhanced with the ability to include some pre-defined variables that refer to common portions of information (for example, the IP address of the LAN interface or the name of the WAN interface) that Cisco Kinetic GMM substitutes into the configuration dynamically for each gateway.

This feature is very flexible, and while care needs to be exercised to prevent configuration issues, it can be used to implement firewall policies (and many more advanced functions). The advanced templates are written in Freemarker markup (https://freemarker.apache.org/). A list of the currently available predefined variables that can be used in the custom advanced templates is shown in Table 1.

Table 1 Currently Available Predefined Variables

Variable Name	Description
gw.sn	GW Serial Number
gw.model	GW Model
gw.wan_if	GW's WAN interface (e.g., "GigabitEthernet0", "Cellular0")
gw.wan_if_sec	GW's Secondary WAN interface (e.g. Dual LTE)
gw.subnet	Subnet for GW's 32 IPs (a /27 address, e.g., "10.9.18.32")
gw.netmask	GW's 32 IP's subnet ("255.255.255.224")
gw.ip	GW's IP (e.g., "10.9.18.33")
gw.ip_prefix	GW IP's first 3 numbers, separated by "." (this makes calculating IPs easier, e.g., "10.9.18")
gw.ip_suffix	GW IP's last byte (e.g., "33")
gw.gos_ip	GW's GuestOS IP (e.g., "10.9.18.34")
gw.lan_if	GW's LAN interface name (e.g., "Gi1", "Vlan1", depending on model)
gw.lan_ip	GW's LAN IP
gw.lan_subnet	GW's LAN subnet
gw.lan_netmask	GW's LAN netmask (e.g., "255.255.255.240")

Table 1 Currently Available Predefined Variables (continued)

Variable Name	Description
gw.lan_wildcard	GW's LAN wildcard (negative of lan_netmask for ACL, e.g., "0.0.0.15")
gw.vpn.pri.ip	Site-to-Site VPN's peer IP (primary)
gw.vpn.sec.ip	Site-to-Site VPN's peer IP (secondary)

A collection of scripts and Advanced Templates have been posted to GitHub. This regularly updated repository contains many of the examples shown throughout this CVD and more. Scripts are available that will easily add the recommended templates for select use cases to your Kinetic GMM organization. Refer to:
https://github.com/CiscoDevNet/iot-gateway-management

Network Design Options

There are several options for connecting devices to the gateway and managing them remotely, all configured using templates in Cisco Kinetic GMM. The Cisco Kinetic GMM product documentation guides you through the configuration of these individual features, but the purpose of this section is to provide examples of how those features can be used together for common use cases.

The options (use cases) are listed in no particular order and care should be taken in reviewing company requirements to determine the best option to meet customer needs, considering several factors including security, complexity, cost, and legacy equipment. To help guide your decision in choosing the best option for your specific use case, use the following questions to begin the discussion:

- Do the remote gateways (and the devices behind them) require access to any resources inside the enterprise intranet or only cloud-based resources on the internet?
- What kind of connectivity is required to the devices located behind the remote gateways? Full reachability to a known IP address or just a specific application/port number on the device?
- What scale of deployment is required (now and in the future)? Would a "cookie-cutter" addressing scheme be beneficial?

Table 2 Available Network Design Options

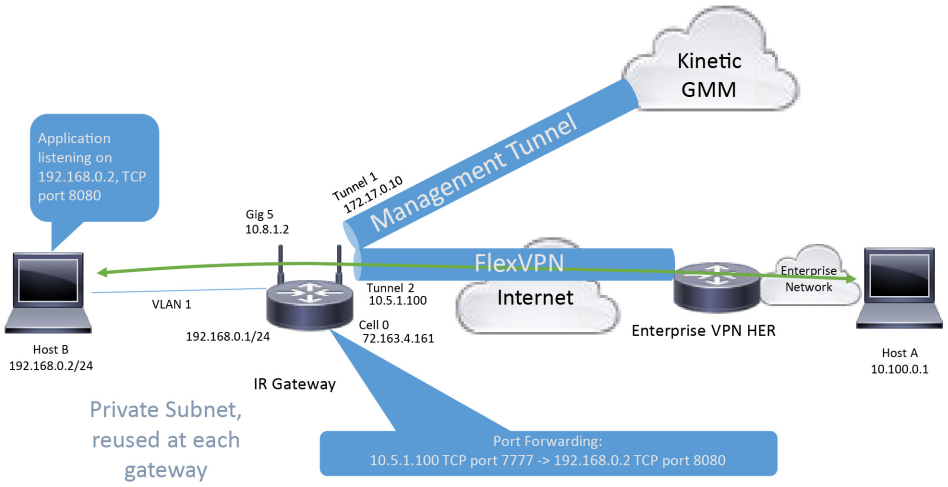
	Management Tunnel	Customer Tunnel	Common Custom LAN Subnet	Unique Custom LAN Subnet	Enterprise Routable LAN Subnet	LAN Subnet with NAT	Port Forwarding
Enterprise VPN with Common Subnet—Option A	X	X	X			X	X
Enterprise VPN with IT-Defined Subnet —Address Overlap—Option B	X	X		X	X		
Enterprise VPN with IT-Defined Subnet—No Address Overlap—Option C	X	X		X	X		
Internet with Common Subnet—Option D	X		X			X	X (advanced template required)
Internet with GMM Assigned Subnet—Option E	X						

Table 3 Remote Site Design Option Decision Matrix

	Price	Remote Site Complexity	Central Site Integration Complexity	Security	Ease of Access to Remote Devices
Option A	Better	Better	Good	Better	Good
Option B	Good	Good	Better	Best	Best
Option C	Good	Good	Better	Better	Best
Option D	Best	Better	n/a	Good	Better
Option E	Best	Best	n/a	Good	Good

Enterprise VPN with Common Subnet—Option A

Figure 2 Enterprise VPN with Common Subnet—Option A



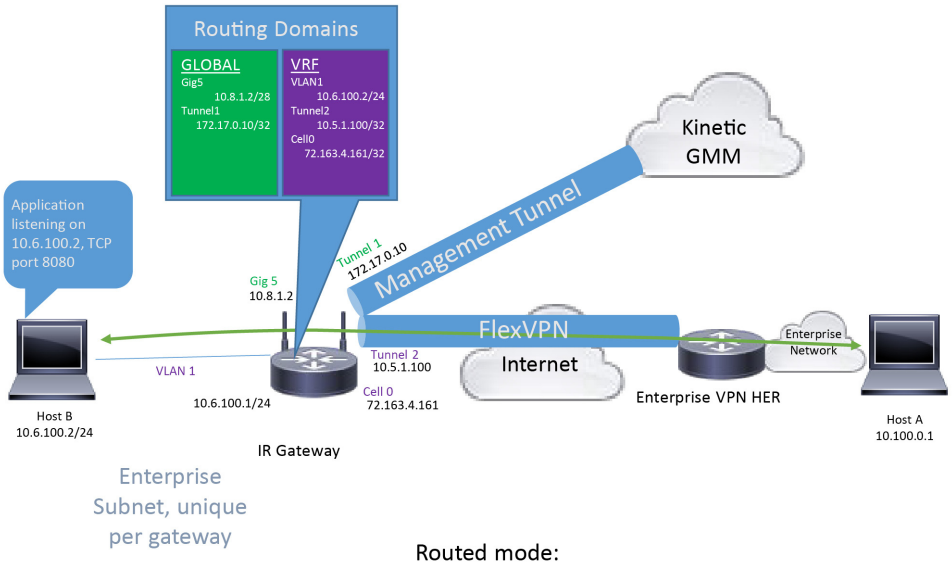
NAT mode:
Host A communicates with 10.5.1.100 on TCP port 7777. The gateway forwards this to Host B at 192.168.0.2 on TCP port 8080.

257145

Option A provides a secure, replicable network design that would be ideal for an enterprise deployment of many remote sites. In this option, the remote sites all use the same LAN subnet, which can be beneficial if there are many cookie cutter remote sites all with similar devices and networks behind the gateway. Port forwarding is enabled to allow remote access (inbound) from the enterprise, across the FlexVPN, and into the network behind the gateway. This option would provide the remote site access to enterprise resources, with optional default routing over the FlexVPN tunnel or split tunneling for outbound traffic from the gateway. This provides a good mix of simplicity and security.

Enterprise VPN with IT-Defined Subnet—Address Overlap—Option B

Figure 3 Enterprise VPN with IT-Defined Subnet—Address Overlap—Option B



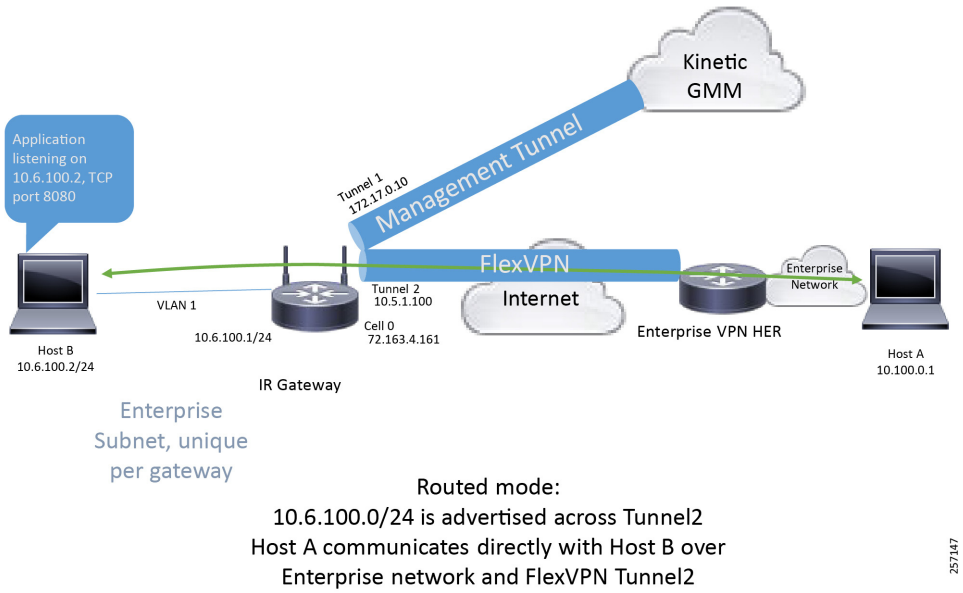
Routed mode:
10.6.100.0/24 is advertised across Tunnel2.
Host A communicates directly with Host B over Enterprise network and FlexVPN Tunnel2.

257146

Option B utilizes routing over the FlexVPN tunnel to the enterprise VPN headend. This means that the 10.6.100.0/24 network is advertised over the Tunnel2 interface, making that network directly reachable (without NAT/Port Forwarding) from the enterprise (Host A). Enterprise routes (including a default route) are similarly advertised over the FlexVPN tunnel causing all data plane traffic (not management) to traverse the tunnel. Since the data plane traffic all goes through the enterprise headend, it can be subject to additional security applications and policies, providing maximum security. The addition of VRF routing domains means that the enterprise IP address space can potentially overlap with the Cisco Kinetic GMM IP address space (both RFC1918 networks). The routing and security provided in this option help make the remote sites as integrated as possible with the main enterprise network.

Enterprise VPN with IT-Defined Subnet—No Address Overlap—Option C

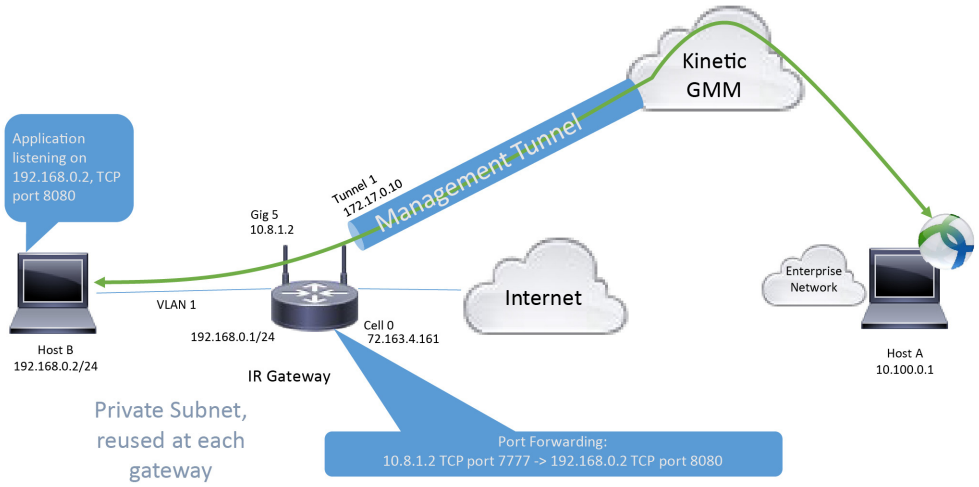
Figure 4 Enterprise VPN with IT-Defined Subnet—No Address Overlap—Option C



Option C is similar to Option B, but without the use of VRFs to separate the Cisco Kinetic GMM and enterprise routing domains. This option is sufficient if there is no chance of the two routing domains having overlapping IP address spaces.

Internet with Common Subnet–Option D

Figure 5 Internet with Common Subnet–Option D



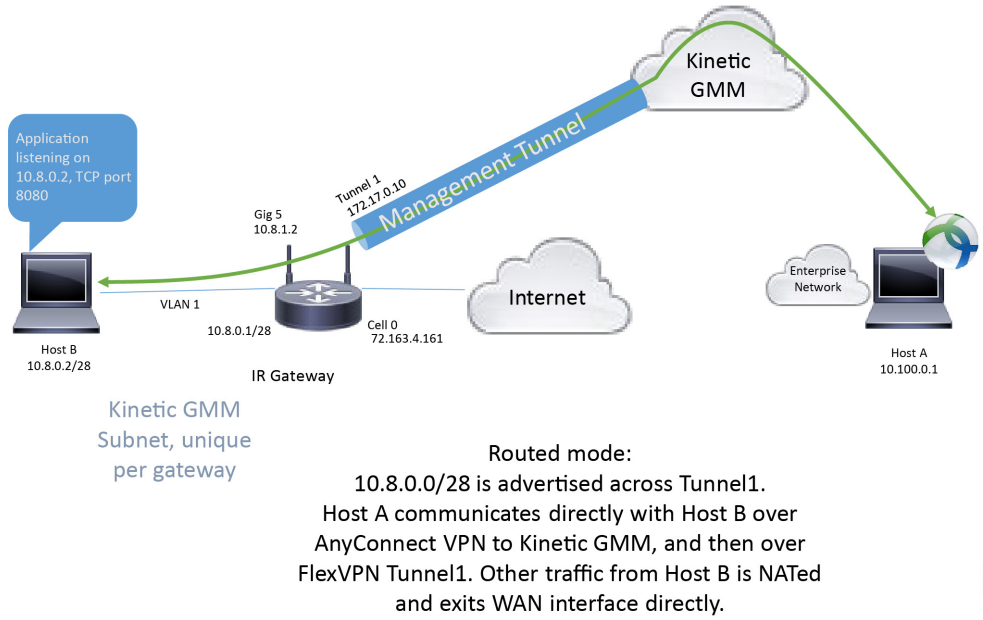
NAT mode:
Host A communicates with 10.8.1.2 on TCP port 7777. The gateway forwards this to Host B at 192.168.0.2 on TCP port 8080.

257148

Option D provides a viable solution for companies that do not want (at least initially) to take on the added expense and complexities of managing a VPN headend and accept that all data plane traffic will traverse the internet directly (not encrypted in a VPN). The use of a custom private subnet that is identical at each remote site enables a “cookie-cutter” installation at each site. This can help simplify deployments when each site will look the same. The addition of port forwarding statements (added with Custom Configs) allows remote users to connect to specific applications or services running behind the gateway by using their AnyConnect client to go through the Cisco Kinetic GMM management tunnel.

Internet with GMM Assigned Subnet—Option E

Figure 6 Internet with GMM Assigned Subnet—Option E



Option E is similar to Option D, except LAN devices are addressed with Cisco Kinetic GMM assigned subnets (unique per site). This change allows the LAN devices to be directly reachable via the Cisco Kinetic GMM management tunnel (and AnyConnect) without the need for NAT/Port Forwarding.

Design Considerations

Connectivity

No matter which method you decide to use for allocating the subnet for the remote site LAN, Cisco Kinetic GMM will configure the gateway to act as a DHCP server for that subnet. The DHCP server functionality will dynamically assign IP addresses (and other information) to any device that requests an IP address in the subnet. Alternatively, you can manually configure devices behind the gateway with a static IP address. It may be desirable to statically configure the IP addresses of LAN devices behind the gateway. This ensures that the port-forwarding rules will consistently work even if there is a site-wide power outage that would otherwise cause DHCP assigned addresses to potentially be re-assigned to different devices.

Security

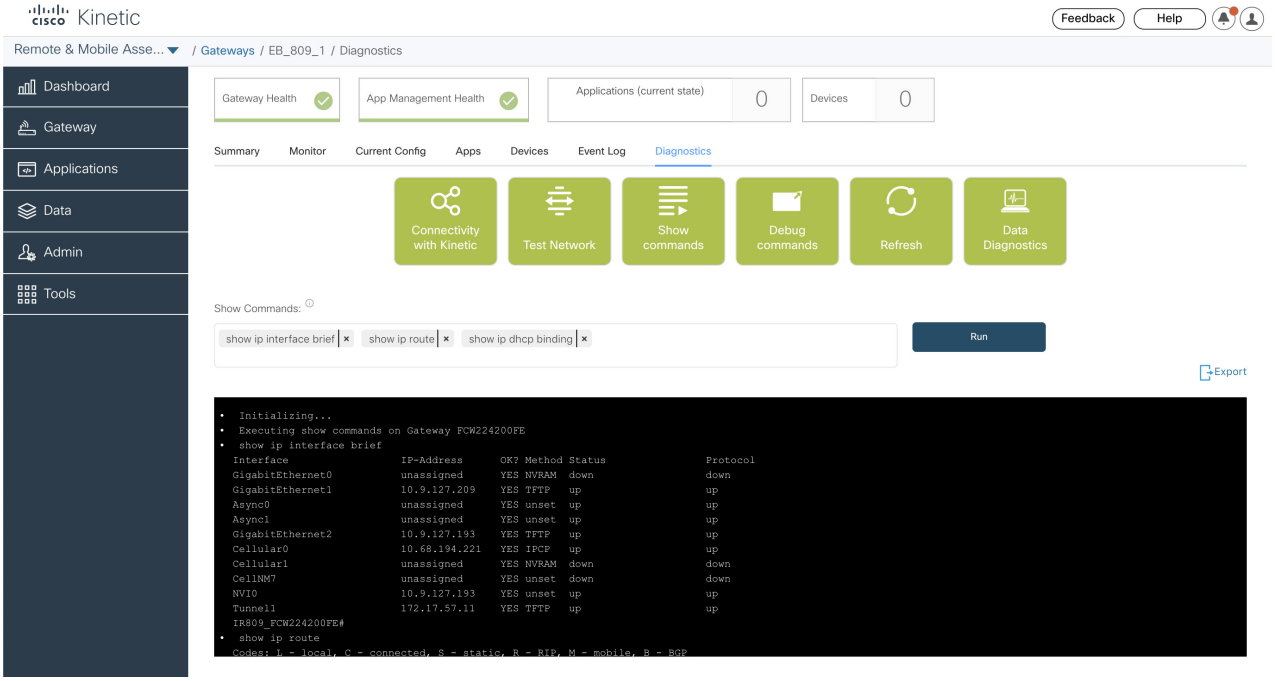
Enabling a basic firewall on the managed gateway can help protect end devices from malicious traffic. However, by utilizing the Cisco Kinetic GMM Custom Config feature within a Template, a firewall configuration consisting of one (or more) access lists whitelisting only the required networks and protocols can be applied to the relevant interfaces. This further locks down access to and from the devices.

Device Management

It is often useful to have an easy method to determine how end devices are connected to the router, specifically to find out what IP address has been dynamically assigned to the device by the gateway. Cisco Kinetic GMM offers a simple way to detect connected devices that are using the MQTT protocol. Once detected, Cisco Kinetic GMM displays basic information about the device including the assigned IP address and MAC address.

Alternatively, for devices that do not support MQTT, the built-in “Show commands” function in Cisco Kinetic GMM can display the IP address of the LAN interface. From there, you can generally increment the last octet to find the next address in the subnet (the DHCP server assigns addresses in order from low to high).

Figure 7 Example of “Show Commands” Diagnostics



Complexity

When selecting the best network design for a specific deployment, it is important to factor in the complexity in terms of effort required to provision the central and remote sites and the expertise required to build and maintain the network. While the use of Cisco Kinetic GMM for management of the remote site gateways greatly simplifies the provisioning and monitoring of those sites, it does not, for example, address the management of the VPN headend.

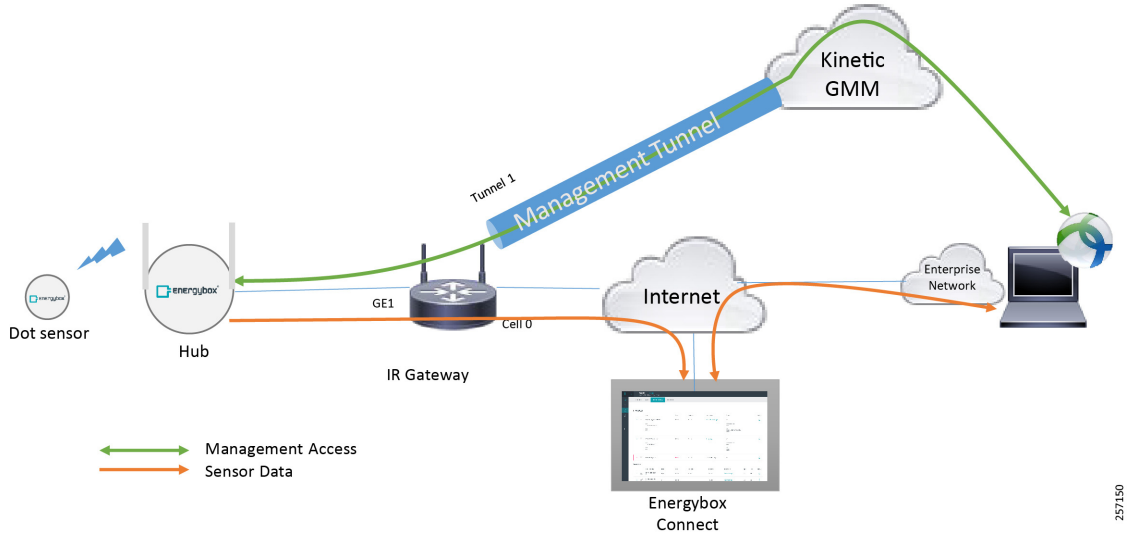
Best Practices

- When remote access to devices behind the router is important, using static IP addressing may be ideal.
- Using FlexVPN with VRF (options A and C) provides the most security for data plane traffic.
- When using static IP addressing for LAN devices behind the managed gateway, only assign IP addresses that are within the address range of “excluded addresses” (configurable for custom subnets).
- It is recommended that if you use DHCP for initial address assignment, power up and/or connect the LAN devices in the order that you want the addresses to be assigned, as the addresses are assigned sequentially.
- Disable unused LAN ports, which helps prevent a malicious person or device from connecting a wired device to the gateway. This can be configured directly in the Cisco Kinetic GMM template.

Implementation

The following two sections show examples of how two of the network design options (D and E) could be implemented for a specific use case involving remote sites that need to be monitored for various environmental conditions (temperature, humidity, etc). In this case, a Cisco 809 Industrial Integrated Services Router (Cisco IR809) gateway was chosen since a single Ethernet connection was required for the connected devices. To monitor environmental conditions, Energybox was selected to provide the sensors and monitoring software to view the sensor data. In each example, a single device (the Energybox hub) is connected directly to the Cisco IR809 gateway. The Energybox sensors communicate wirelessly with the Energybox hub and the data is subsequently forwarded through the Cisco IR809 to the Energybox Connect cloud monitoring application. These examples also require remote access to the Energybox hub and do not utilize an enterprise FlexVPN since the destination application is in the cloud.

Figure 8 Cisco IR809 with Energybox Hub and Sensors

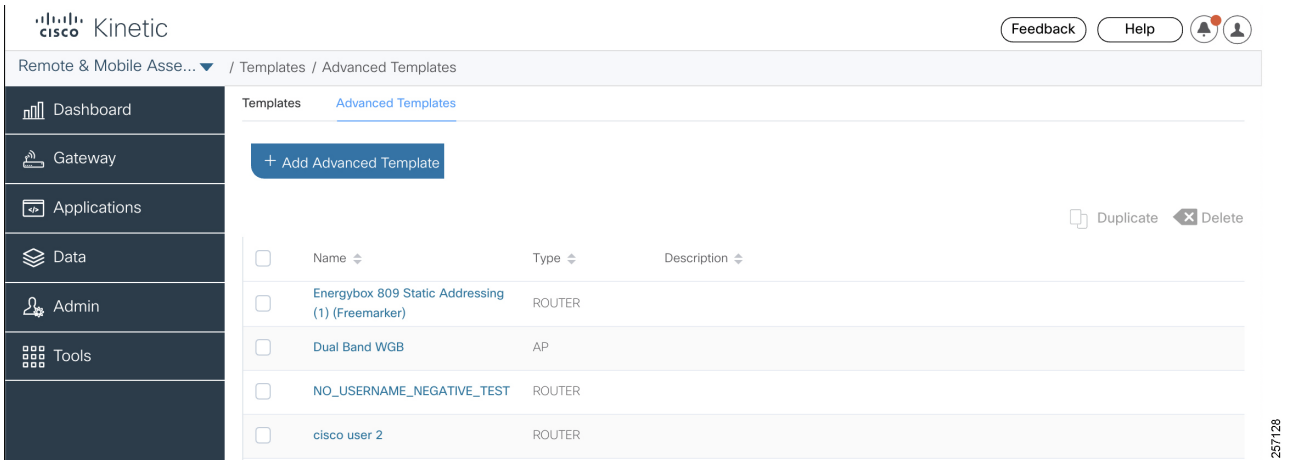


For additional information on Energybox, see: https://www.energybox.com

Implementation of Option D with Cisco IR809 and Energybox

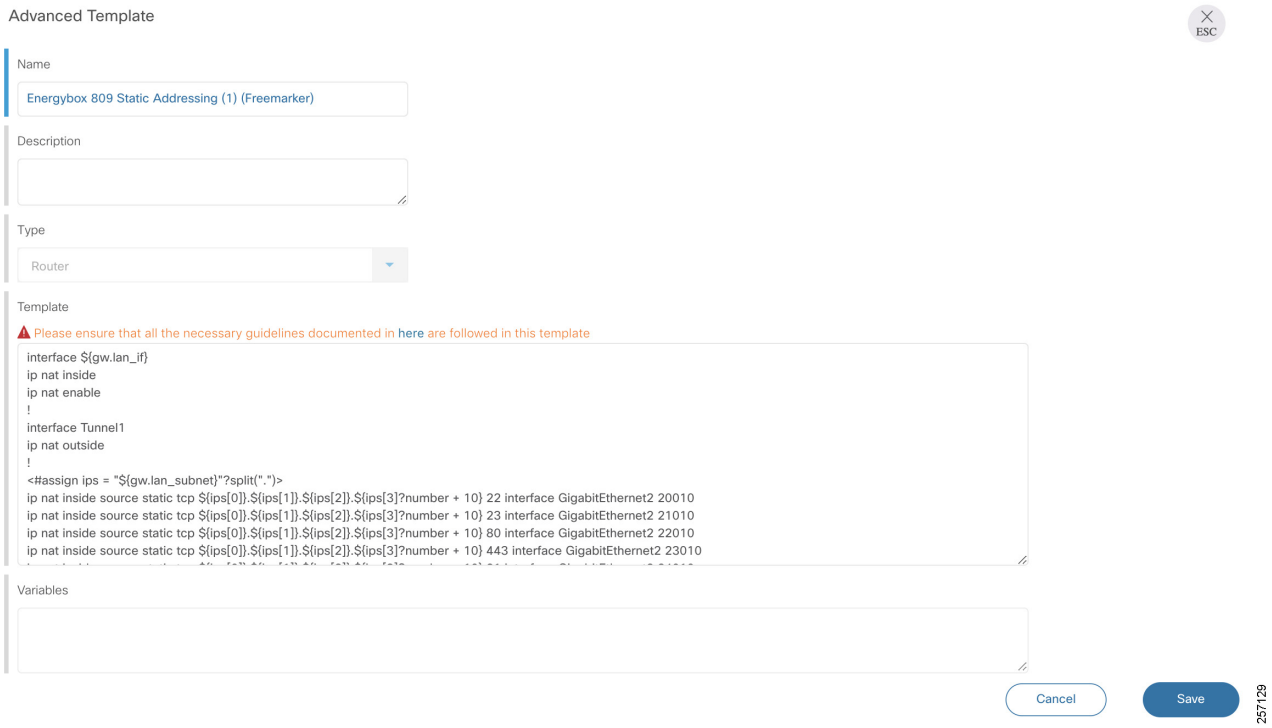
As an administrator, log into a Cisco Kinetic GMM org that has the Advanced Template feature enabled (this can be requested using **Help** > **Contact Us**). Browse to the Advanced Template page shown in Figure 9 and click **Add Configuration**.

Figure 9 Implementation of Option D—Add Custom Configuration



In the Advanced Template page, enter a name for the advanced template and copy/paste the configuration commands shown after the screenshot.

Figure 10 Implementation of Option D—Advanced Template Detail



The example configuration below configures a series of port-forwarding statements for ten devices behind the gateway, each with NAT statements that forward traffic for several common protocols (Telnet, SSH, HTTP, SSL, FTP). Next an access list, “LAN_ACL”, is created which whitelists the required protocols and ports necessary for an Energybox hub to operate correctly. The access list is then applied on the LAN interface in the inbound direction. The configuration could be modified to permit other ports and protocols or to support more or fewer end devices.

```
interface ${gw.lan_if}
ip nat inside
ip nat enable
!
interface Tunnel1
ip nat outside
!
<#assign ips = "${gw.lan_subnet}"?split(".")>
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 10} 22 interface
GigabitEthernet2 20010
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 10} 23 interface
GigabitEthernet2 21010
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 10} 80 interface
GigabitEthernet2 22010
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 10} 443 interface
GigabitEthernet2 23010
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 10} 21 interface
GigabitEthernet2 24010

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 11} 22 interface
GigabitEthernet2 20011
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 11} 23 interface
GigabitEthernet2 21011
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 11} 80 interface
GigabitEthernet2 22011
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 11} 443 interface
GigabitEthernet2 23011
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 11} 21 interface
GigabitEthernet2 24011

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 12} 22 interface
GigabitEthernet2 20012
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 12} 23 interface
GigabitEthernet2 21012
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 12} 80 interface
GigabitEthernet2 22012
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 12} 443 interface
GigabitEthernet2 23012
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 12} 21 interface
GigabitEthernet2 24012

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 13} 22 interface
GigabitEthernet2 20013
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 13} 23 interface
GigabitEthernet2 21013
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 13} 80 interface
GigabitEthernet2 22013
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 13} 443 interface
GigabitEthernet2 23013
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 13} 21 interface
GigabitEthernet2 24013

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 14} 22 interface
GigabitEthernet2 20014
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 14} 23 interface
GigabitEthernet2 21014
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 14} 80 interface
GigabitEthernet2 22014
```

```
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 14} 443 interface
GigabitEthernet2 23014
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 14} 21 interface
GigabitEthernet2 24014

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 15} 22 interface
GigabitEthernet2 20015
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 15} 23 interface
GigabitEthernet2 21015
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 15} 80 interface
GigabitEthernet2 22015
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 15} 443 interface
GigabitEthernet2 23015
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 15} 21 interface
GigabitEthernet2 24015

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 16} 22 interface
GigabitEthernet2 20016
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 16} 23 interface
GigabitEthernet2 21016
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 16} 80 interface
GigabitEthernet2 22016
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 16} 443 interface
GigabitEthernet2 23016
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 16} 21 interface
GigabitEthernet2 24016

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 17} 22 interface
GigabitEthernet2 20017
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 17} 23 interface
GigabitEthernet2 21017
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 17} 80 interface
GigabitEthernet2 22017
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 17} 443 interface
GigabitEthernet2 23017
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 17} 21 interface
GigabitEthernet2 24017

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 18} 22 interface
GigabitEthernet2 20018
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 18} 23 interface
GigabitEthernet2 21018
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 18} 80 interface
GigabitEthernet2 22018
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 18} 443 interface
GigabitEthernet2 23018
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 18} 21 interface
GigabitEthernet2 24018

ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 19} 22 interface
GigabitEthernet2 20019
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 19} 23 interface
GigabitEthernet2 21019
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 19} 80 interface
GigabitEthernet2 22019
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 19} 443 interface
GigabitEthernet2 23019
ip nat inside source static tcp ${ips[0]}.${ips[1]}.${ips[2]}.${ips[3]?number + 19} 21 interface
GigabitEthernet2 24019

!
ip access-list extended LAN_ACL
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq domain
```

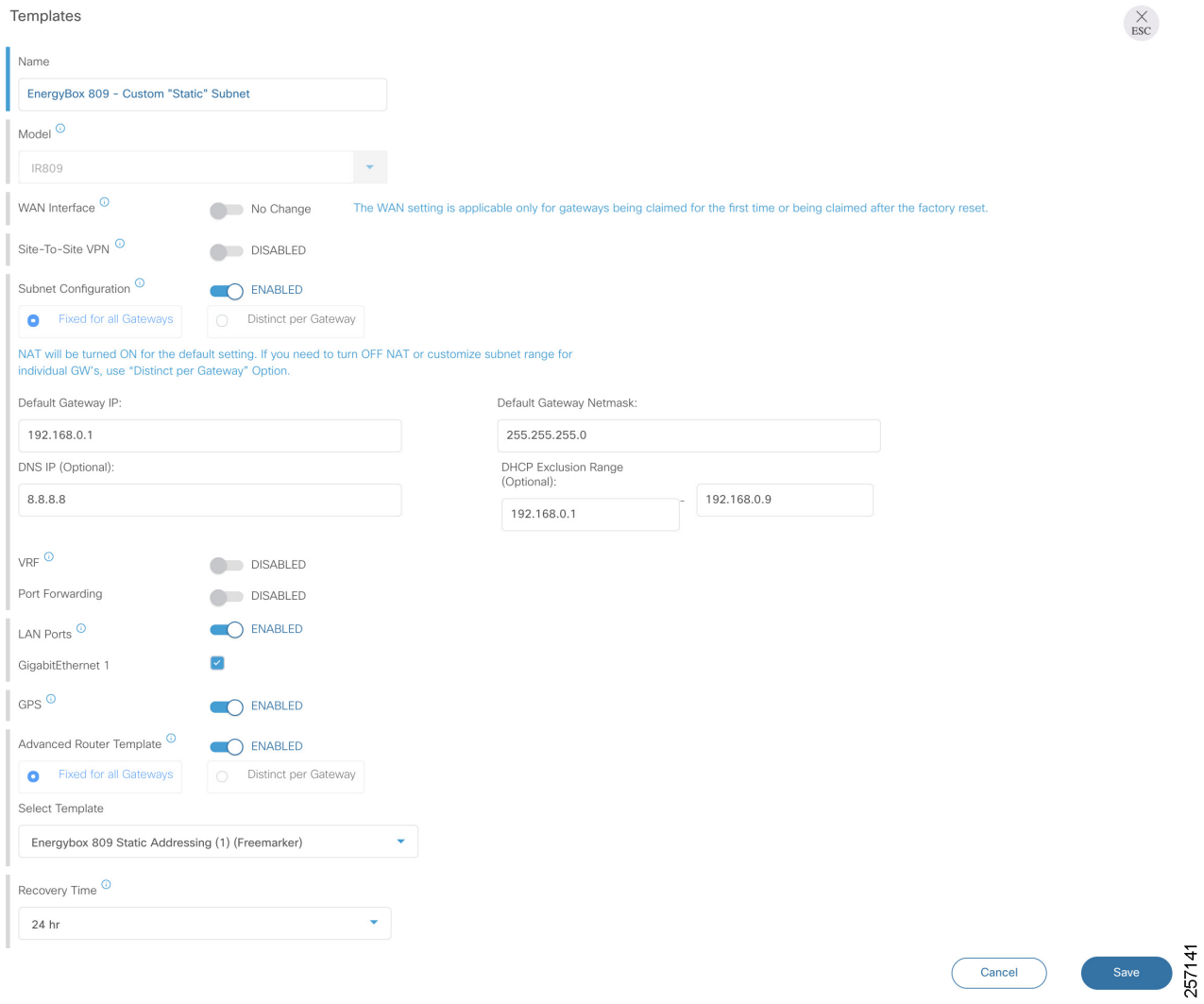
```
 permit udp ${gw.lan_subnet} ${gw.lan_wildcard} any eq ntp
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq www
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq 1883
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} eq www any
 permit udp any eq bootpc any eq bootps
 permit udp ${gw.lan_subnet} ${gw.lan_wildcard} any eq domain
 permit icmp ${gw.lan_subnet} ${gw.lan_wildcard} host ${gw.lan_ip}
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq 443
 deny ip any any
!
interface ${gw.lan_if}
 ip access-group LAN_ACL in
```

Note: If using an EnergyPro to monitor power consumption, add the following line to the “LAN_ACL” access list to permit MODBUS-TCP and streaming data to the Energybox backend:

```
permit tcp {{ gw.lan_subnet }} {{ gw.lan_wildcard }} eq 502 any
permit tcp {{ gw.lan_subnet }} {{ gw.lan_wildcard }} any eq 3772
```

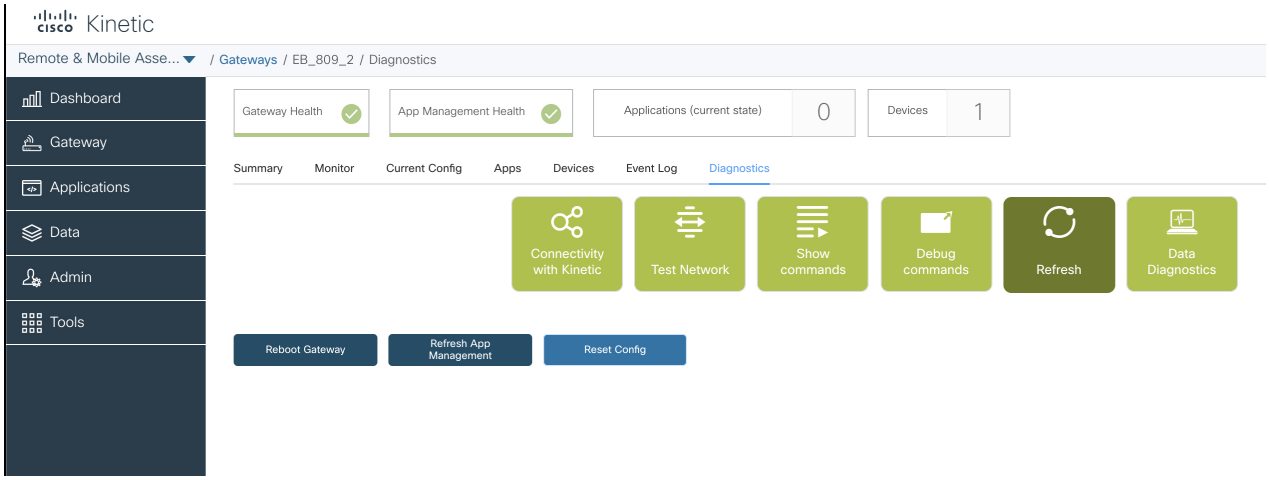
Next, the custom configuration is applied to the existing Template:

Figure 11 Implementation of Option D–Apply Advanced Router Template to Template



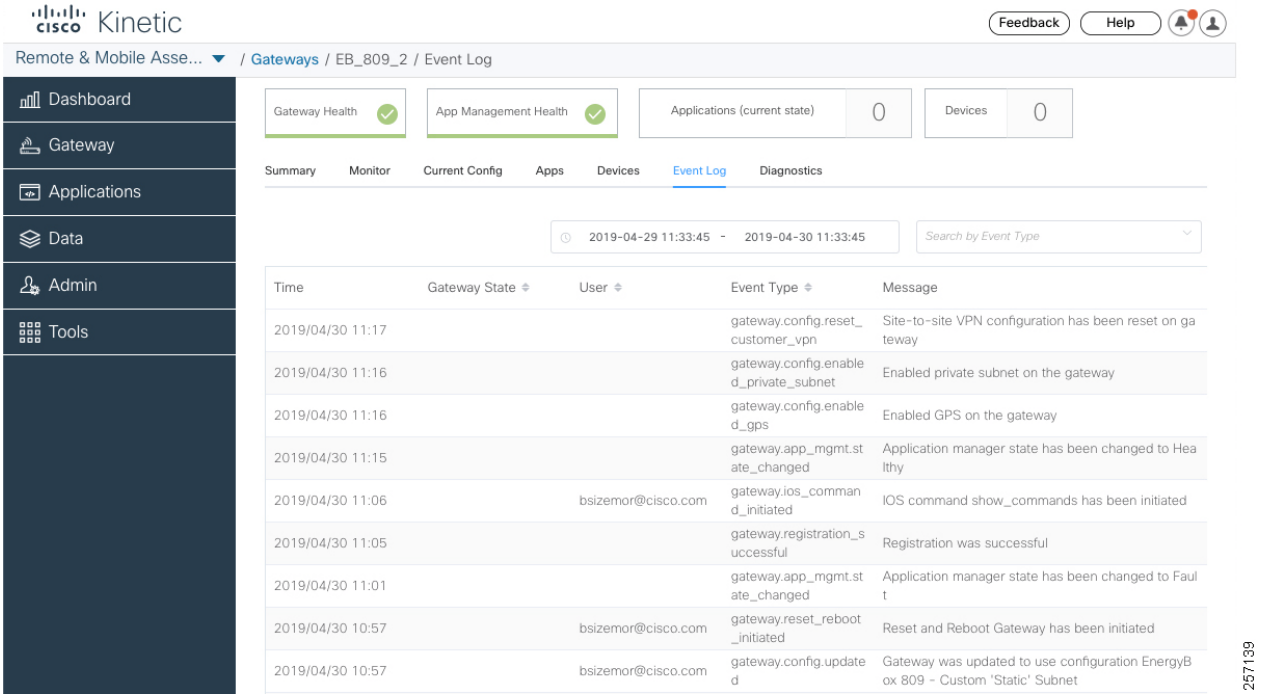
Even though it indicates that the template will be updated on affected gateways, you should use the Reset Config feature for the gateway(s) to effectively apply the Template, including the Advanced Template. After clicking **Reset Config**, the process takes about 20 minutes to complete and basically clears the configuration, reboots the router, and reapplies the entire configuration. The advanced template is the last piece to be applied and it will only be applied if the other pieces of configuration are applied successfully.

Figure 12 Implementation of Option D—Reset Config



Monitoring the Event Log for the gateway shows you the status of the configuration. The process took 20 minutes on a Cisco IR809.

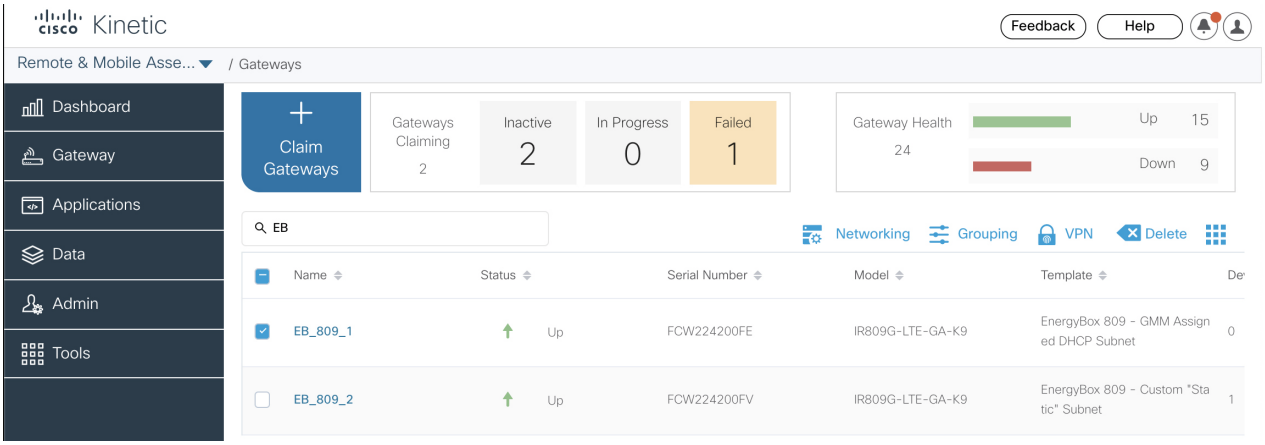
Figure 13 Implementation of Option D— Event Log



After the configuration is complete, port forwarding functionality can be utilized to access the Energybox hub located behind the Cisco IR809. The hub is statically configured with 192.168.0.10 and is reachable locally on port TCP/80, which is NATed to the Cisco Kinetic GMM assigned GigabitEthernet2 IP address (accessible over Mgmt Tunnel1) on port TCP/22010 using HTTP. Connectivity over the management tunnel is provided through the Cisco AnyConnect client running on a users computer that is pointed to the Kinetic GMM management VPN.

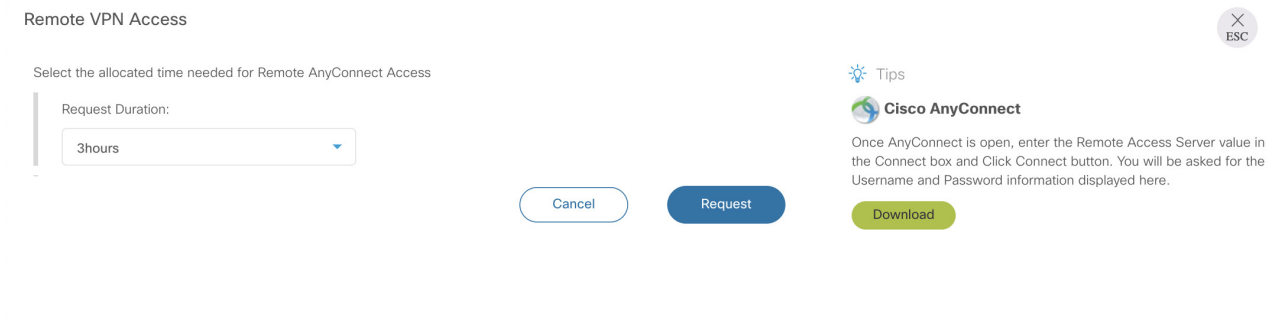
To connect to the Kinetic management VPN, first find the VPN details for your gateway by going to Gateways, then checking the box beside the gateway, and clicking the VPN button at the top.

Figure 14 Select VPN Button for Gateway



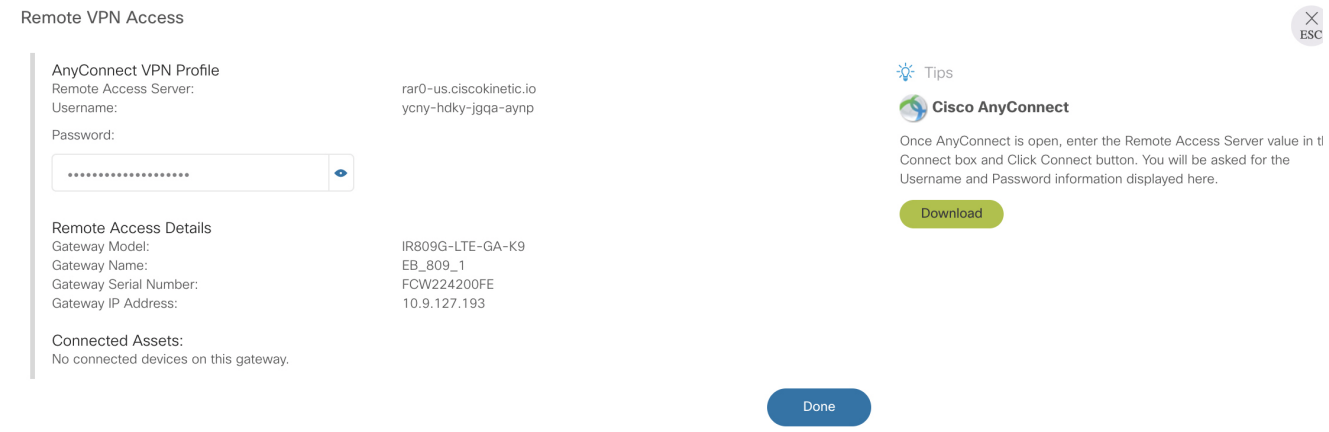
Kinetic GMM will prompt the user to specify a duration during which they will be granted VPN access. After the specified duration expires, the VPN session will be terminated.

Figure 15 Request VPN Access



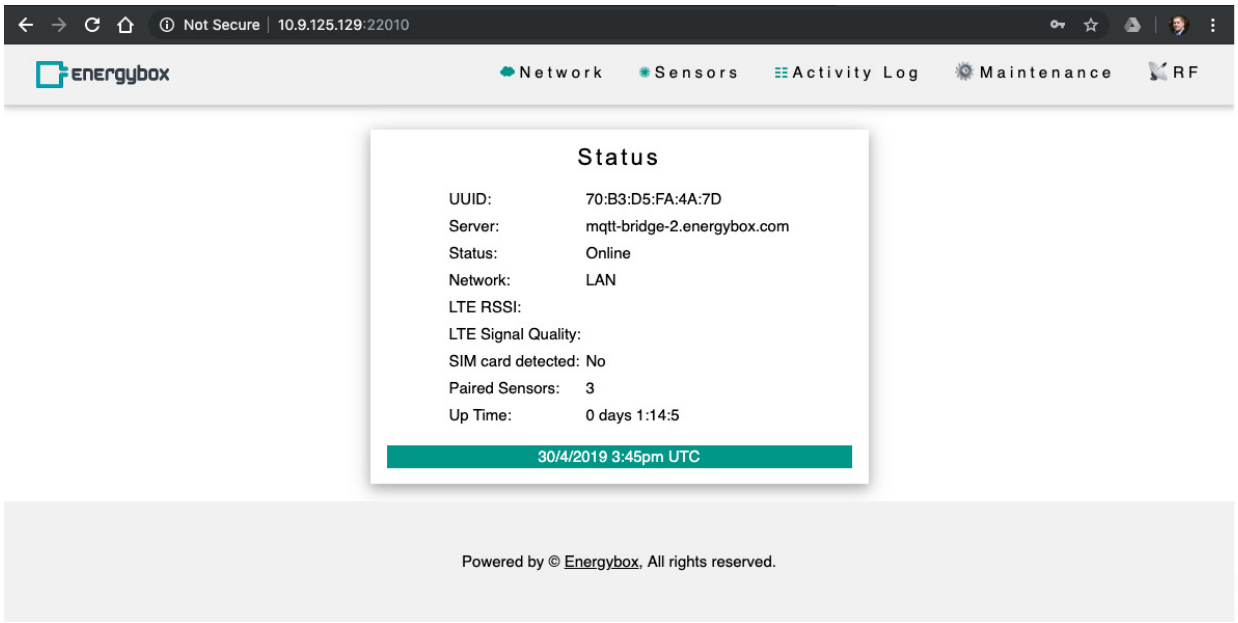
Remote VPN Access details will be displayed that include the Remote Access Server, Username, and Password. Enter this information in AnyConnect to connect to the Kinetic GMM management VPN. Note that this page also shows the time remaining for the VPN session before it is terminated.

Figure 16 Remote Access VPN Details



The LAN subnet on the gateway will be identical for all gateways, which makes it a very scalable and easily replicable solution. The IP address assignments across different sites can be the same. Initially, the addresses can be assigned by DHCP if desired (it is configured this way) and then subsequently changed to be assigned statically on a LAN device-by-device basis. For example, the first device connected behind the gateway will be assigned 192.168.0.10, the next device will be assigned 192.168.0.11, etc. Static addressing is highly recommended because if there is a site-wide power outage, DHCP leases would no longer be maintained and the LAN devices could come up in an unexpected order; therefore the resulting addresses would likely not be the same as before the power outage, thereby breaking the NAT statements which are no longer valid.

Figure 17 Implementation of Option D—Energybox Hub Status



Once connectivity is established for the Cisco gateway and hub, the Energybox data can be seen in the Energybox Connect cloud portal. The set up of the Energybox components is covered in detail in the Energybox documentation.

Figure 18 Implementation of Option D—Energybox Connect Gateways and Sites

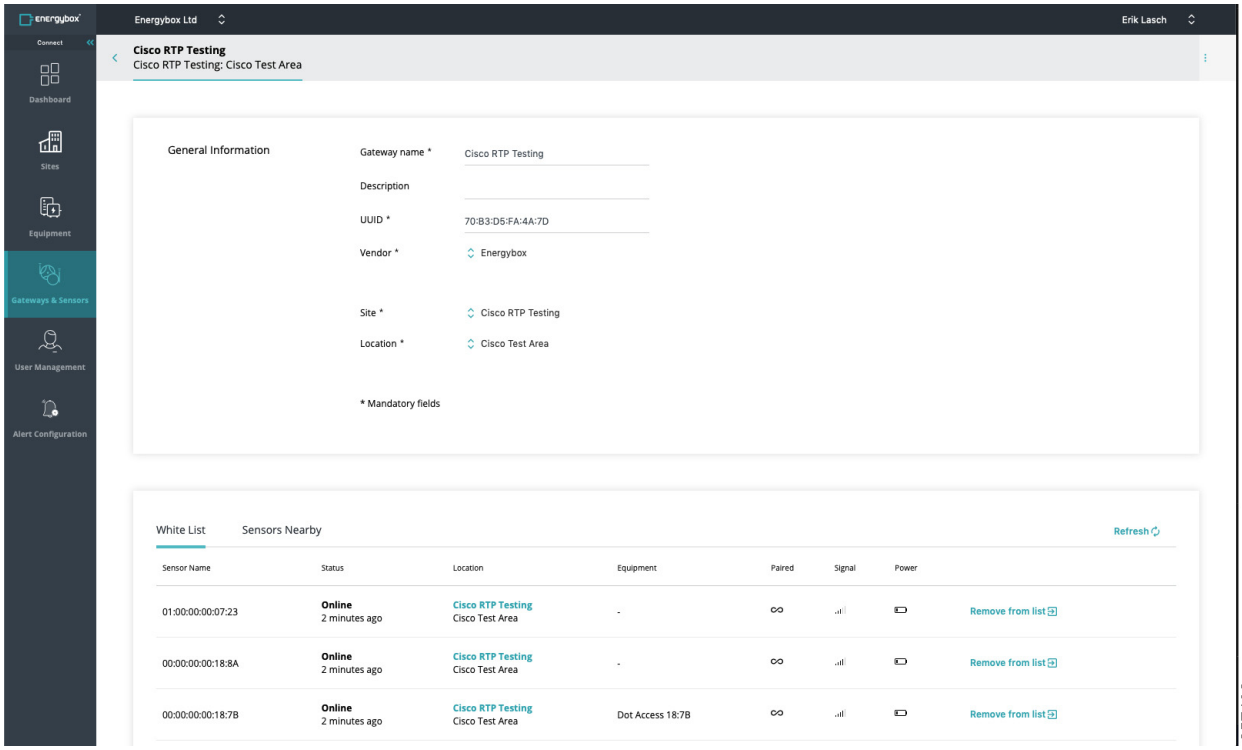


Figure 19 Implementation of Option D—Energybox Connect Sites

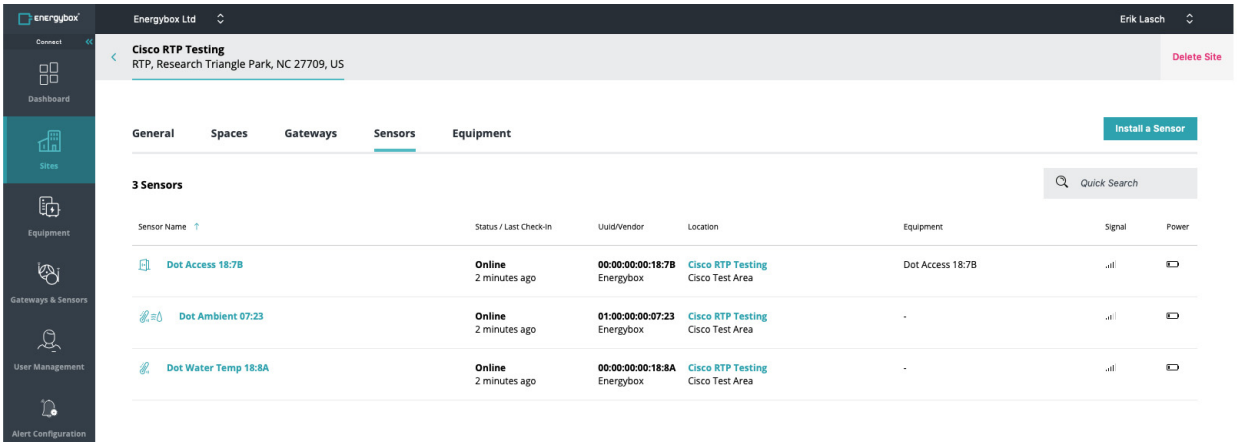
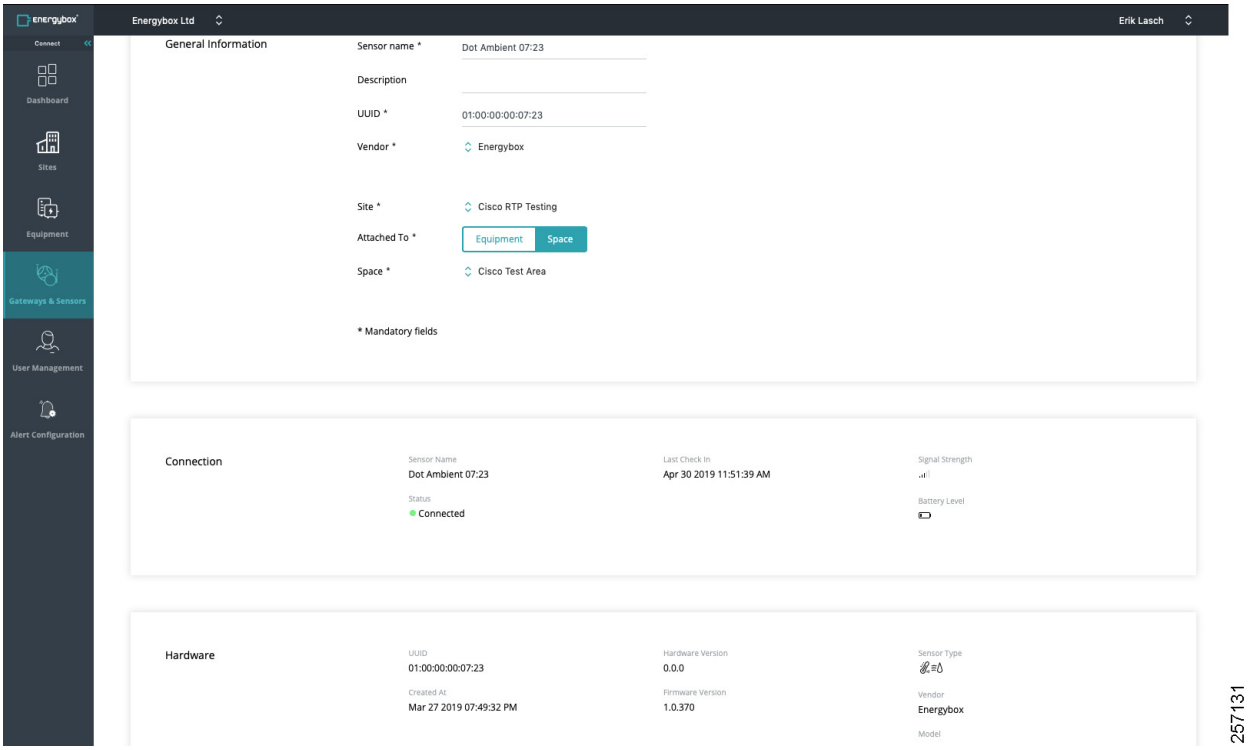


Figure 20 Implementation of Option D—Energybox Connect Site Detail



Implementation of Option E with Cisco IR809 and Energybox

As an administrator, log into a Cisco Kinetic GMM org that has the advanced template feature enabled (this can be requested using **Help** > **Contact Us**). Browse to the Advanced Template page as shown in the first implementation example, click **Add Advanced Template**, and copy/paste the configuration below.

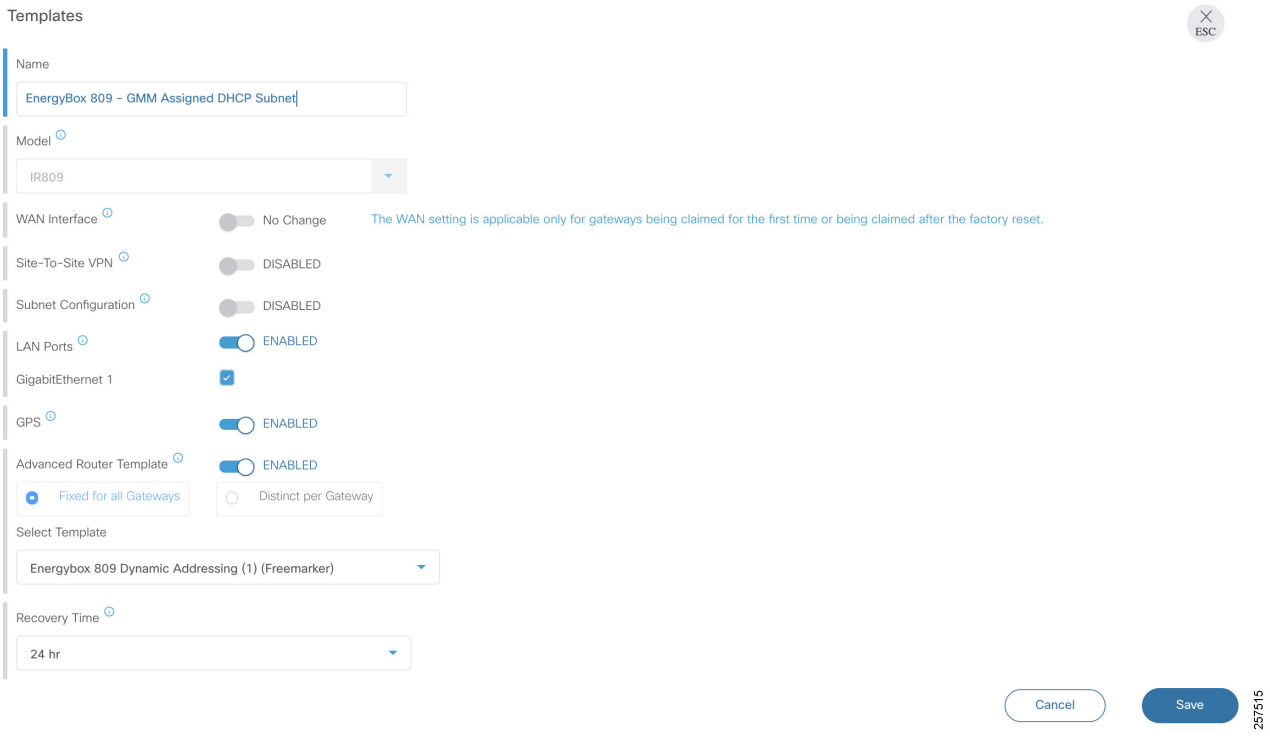
```
!
ip access-list extended LAN_ACL
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq domain
 permit udp ${gw.lan_subnet} ${gw.lan_wildcard} any eq ntp
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq www
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq 1883
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} eq www any
 permit udp any eq bootpc any eq bootps
 permit udp ${gw.lan_subnet} ${gw.lan_wildcard} any eq domain
 permit icmp ${gw.lan_subnet} ${gw.lan_wildcard} host ${gw.lan_ip}
 permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq 443
 deny ip any any
!
interface ${gw.lan_if}
 ip access-group LAN_ACL in
```

Note: If using an EnergyPro to monitor power consumption, add the following line to the **"**LAN_ACL**"** access list to permit MODBUS-TCP and streaming data to the Energybox backend:

```
permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} eq 502 any
permit tcp ${gw.lan_subnet} ${gw.lan_wildcard} any eq 3772
```

Next, apply this to the existing Template:

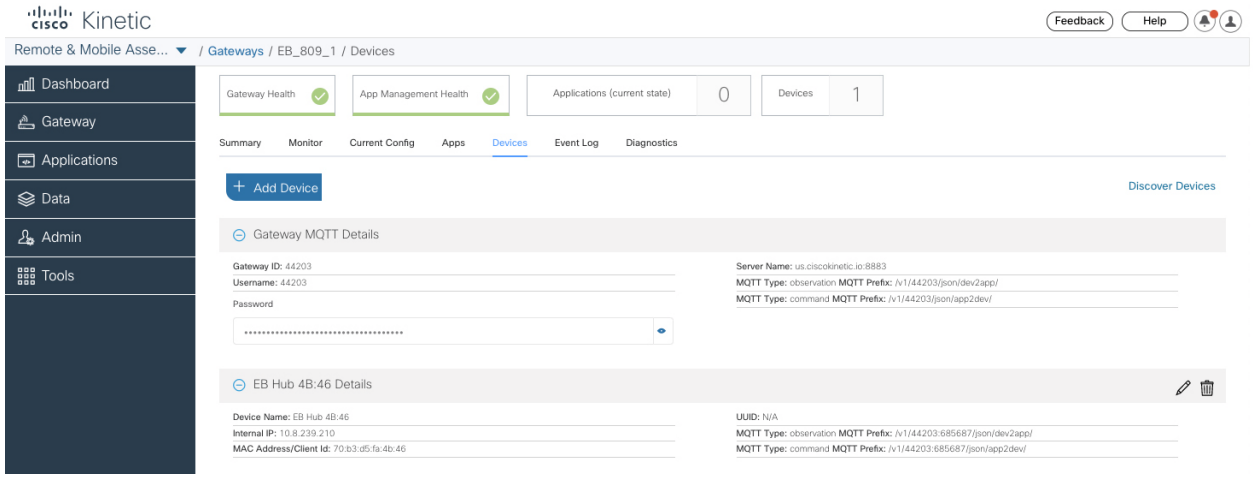
Figure 21 Implementation of Option E—Apply Custom Configuration to Template



Even though it indicates that the template will be updated on affected gateways, you should use the Reset Config feature for the gateway(s) to effectively apply the Template, including the Advanced Template. After clicking **Reset Config**, the process takes about 20 minutes to complete and basically clears the configuration, reboots the router, and reapplies the entire configuration. The Advanced Template is the last piece to be applied and it will only be applied if the other pieces of configuration are applied successfully.

After the configuration is complete, identify the IP address assigned to the Energybox hub. There are two methods to do this without having CLI access to the gateway. The first method in Cisco Kinetic GMM is to use the Device Discovery feature to identify connected MQTT devices behind the gateway that have been assigned an IP address via DHCP. In Figure 22, the **Discover Devices** button has already been clicked and the "EB Hub 4B:46" device (an Energybox hub) has been added. On this screen you can see the IP address (10.8.239.210) that was assigned from the Cisco Kinetic GMM managed subnet to the hub.

Figure 22 Implementation of Option E—Devices



Another method for determining the IP address assigned to the hub is to login to https://connect.energybox.com. From there you will need to add the hub by entering the UUID which is printed on the bottom of the hub. The details of this process are documented in the Energybox documentation. Once the hub has been registered in the Energybox Connect portal and the hub checks in, it will show up in the list of hubs. Included in the Configuration section of this page is the current IP address, which should match that shown by Discovering Devices in Cisco Kinetic GMM.

Figure 23 Implementation of Option E—Energybox Connect Installables

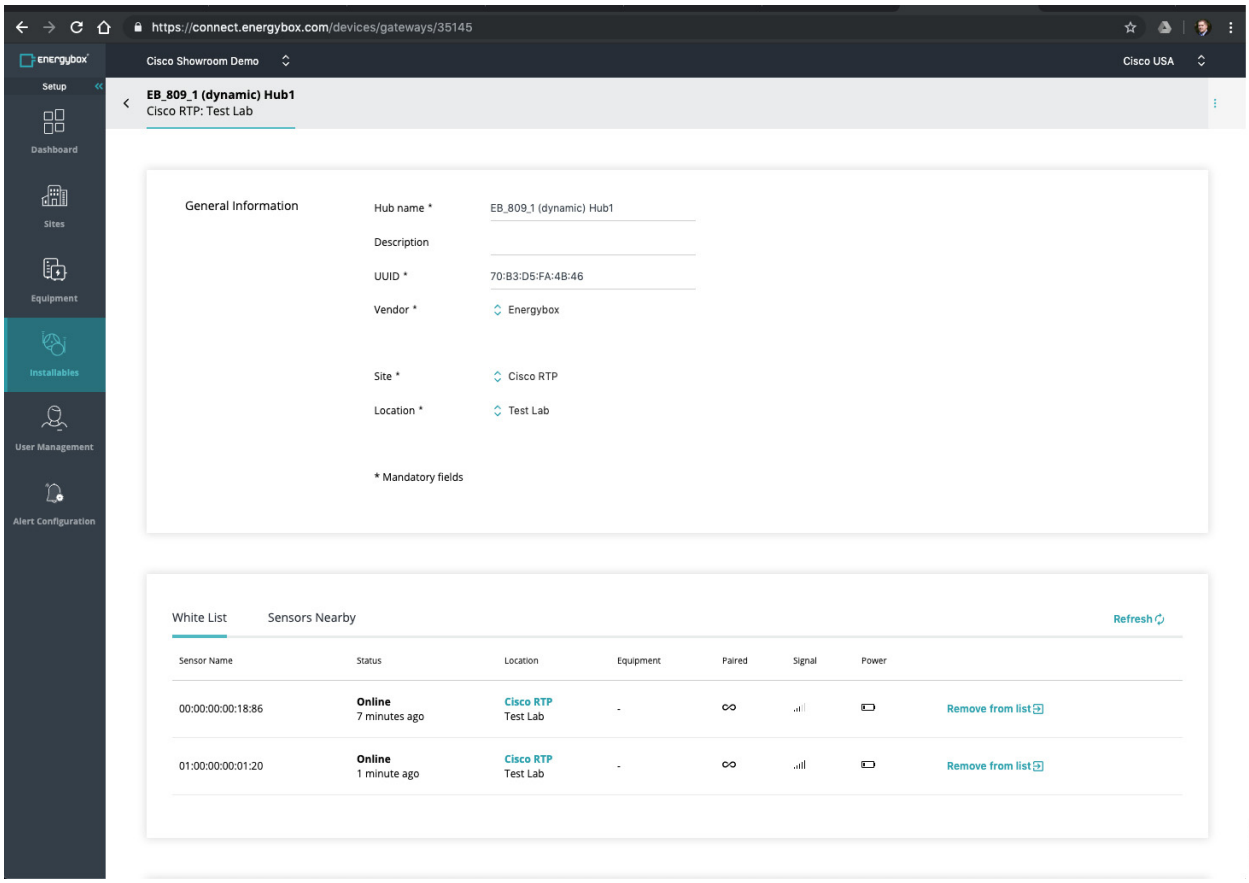
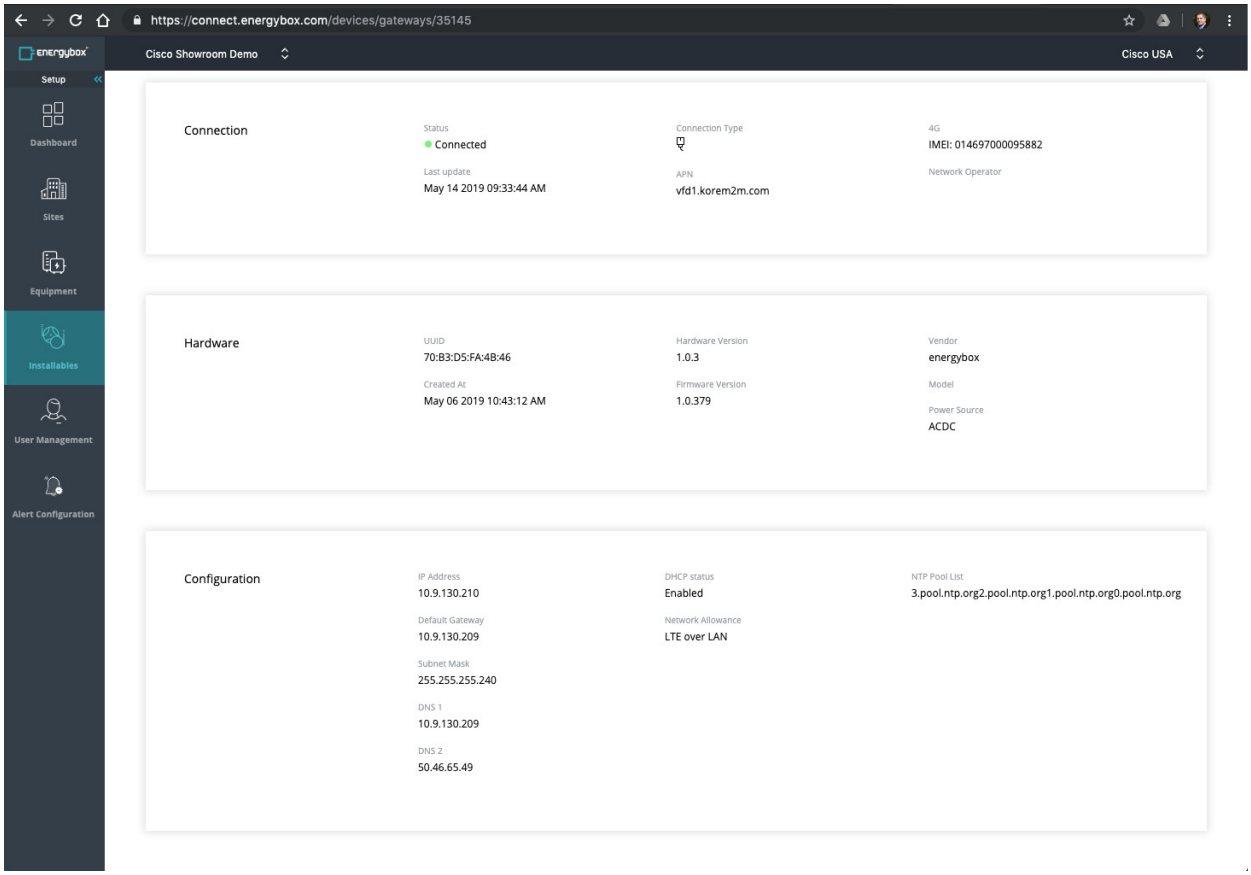


Figure 24 Implementation of Option E—Energybox Connect Hub Details



With this IP address identified, the hub can be accessed directly after logging into the Cisco Kinetic GMM Management VPN using the AnyConnect client. The hub configuration can then be customized to meet your needs using its web interface.

Figure 25 Implementation of Option E–Energybox Hub Network Settings–Static IP

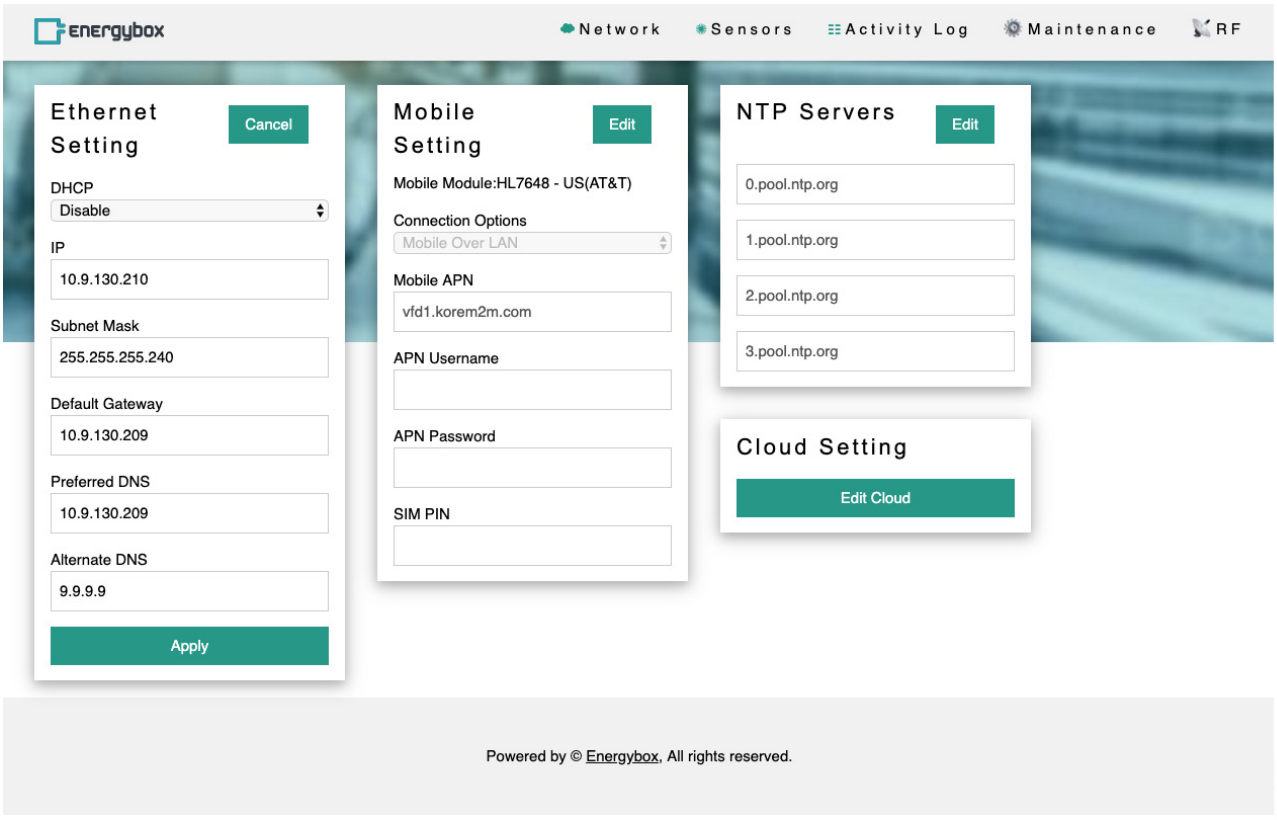


Figure 26 Implementation of Option E—Energybox Hub Network Settings—Dynamic IP

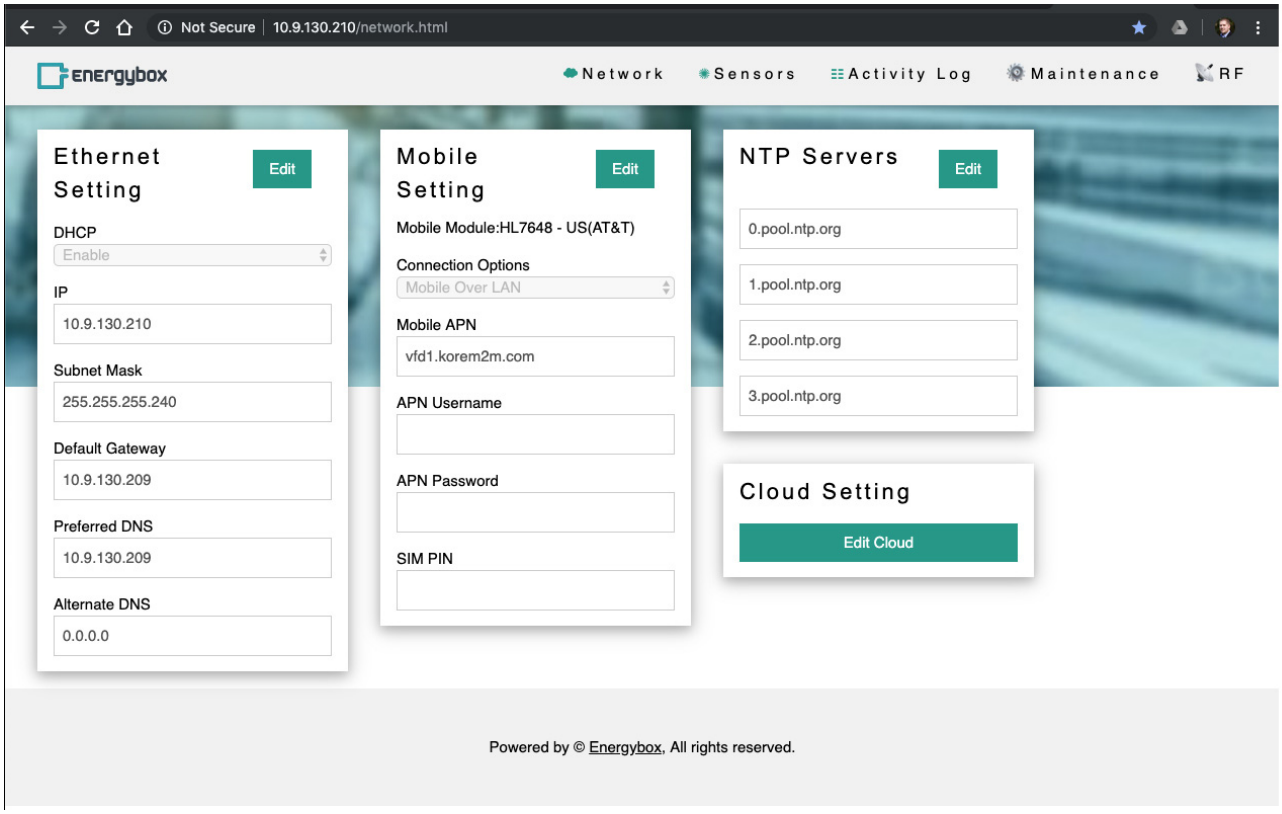
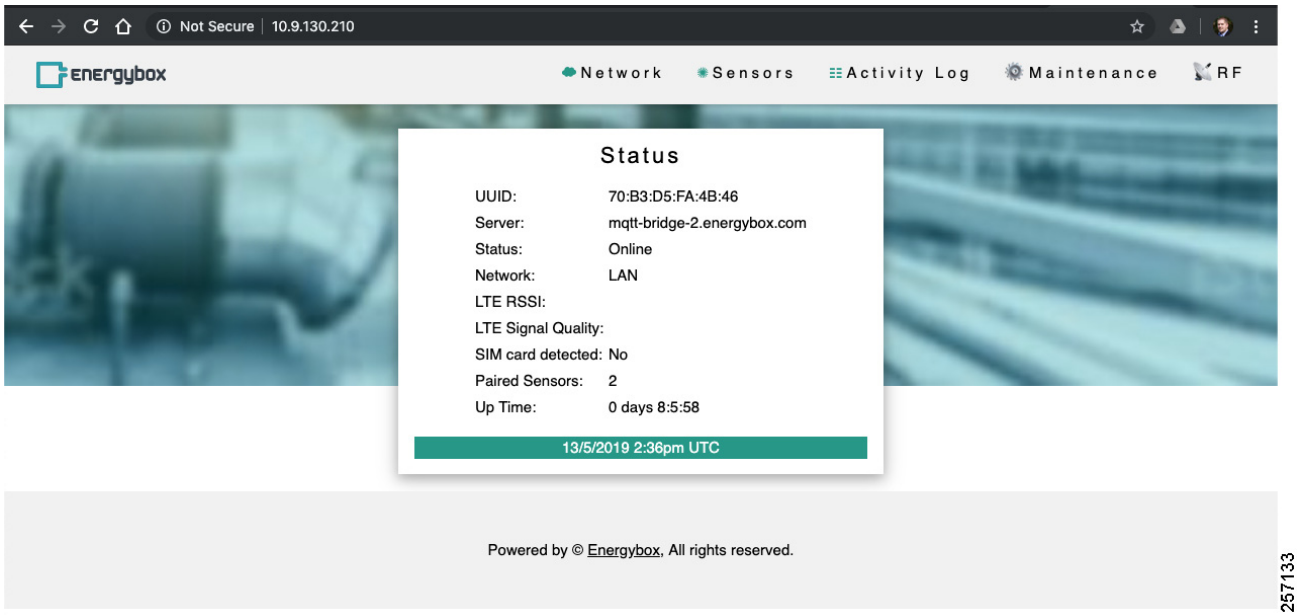


Figure 27 Implementation of Option E—Energybox Hub Status



With the configuration described in this section, the end device behind the gateway (such as an Energybox hub) is dynamically assigned an IP address by the gateway from the Cisco Kinetic GMM managed subnet. If you want the end device to maintain the same IP address in the event of a site-wide power outage or other catastrophic failure, the hub can be configured to use statically assigned addressing. In this case, configure the same (dynamically assigned) address

as a static address, as well as the netmask and default gateway. It should be noted that if the Cisco gateway is ever unclaimed and reclaimed, it will likely be assigned a different subnet from Cisco Kinetic GMM, thereby breaking connectivity to the end device if it was statically assigned an IP address (in a different subnet).

The Energybox sensor data can been viewed in the Energybox Connect portal.

Glossary

Term	Definition
AAA	Authentication, Authorization, and Accounting
AP	Access Point
APN	Access Point Name
AR	Active Router
CAPWAP	Control and Provisioning of Wireless Access Points
CLB	Cluster Load Balancing
CVD	Cisco Validated Design
DMVPN	Dynamic Multipoint VPN
DNS	Domain Name System
DoS	Denial of Service
DPD	Dead Peer Detection
EAP	Extensible Authentication Protocol
EAPoL	EAP over LAN
EEM	Embedded Event Manager
GMM	Cisco Gateway Management Module
GPT	Cisco Kinetic Gateway Provisioning Tool
GRE	Generic Routing Encapsulation
HER	Headend Router
HSPA	High Speed Packet Access
HSRP	Hot Standby Router Protocol
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IKE	Internet Key Exchange
IoT	Internet of Things
IPS	Intrusion Prevention System
IR	Industrial Router
ISAKMP	Internet Security Association and Key Management Protocol
ISE	Cisco Identity Services Engine
LAP	Lightweight Access Point
LLG	Least Loaded Gateway
LTE	Long Term Evolution
LWAP	Lightweight Access Point

Term	Definition
MIMO	Multiple-Input and Multiple-Output
MPLS	Multiprotocol Label Switching
MQC	Modular QoS
mSATA	mini-Serial Advanced Technology Attachment
NAT	Network Address Translation
NGE	Cisco Next-Generation Encryption
NHRP	Next Hop Resolution Protocol
NTP	Network Time Protocol
PoE	Power over Ethernet
PSK	Pre-Shared Keys
RaMA	Cisco Remote and Mobile Assets
RFC	Request for Comments
RHEL	Red Hat Enterprise Linux
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SFP	Small Form-Factor Pluggable
SIM	Subscriber Identification Module
SVI	Switched Virtual Interface
UDP	User Datagram Protocol
VIP	Virtual IP address
VPN	Virtual Private Network
VRF	Virtual Route Forwarding
VTI	Virtual Tunnel Interface
vWLC	virtual Wireless LAN Controller
WAF	Web Application Firewall
WAN	Wide Area Network
WGB	Workgroup Bridge
WLC	Cisco Wireless LAN Controller
ZTD	Zero-Touch Deployment