

# Emergency Calling - 04 - Network Sites

## In This Article

- Overview
- Add a Network Site & Subnets
  - Teams Admin Center
  - PowerShell

# Overview

**Network Sites** are used to determine which of the below Teams policies are assigned to a Teams Client. This is done using corporate network subnets, and is a separate configuration from the Emergency Address network configuration.

- Emergency Calling Policy
- Emergency Call Routing Policy

When a user's device is external (or working from home), and not located in a Network Site, the Teams client will be assigned the **Global (Org-Wide Default)** policies automatically.



## NOTE

Full planning and prerequisite details can be found in the [Emergency Calling - Overview and Planning](#) article.

## Add a Network Site & Subnets



## NOTE

A Network Region is required when creating the first Network Site.

## Teams Admin Center

In the Teams admin center, browse to: **Locations > Network Topology > Network Sites** (tab)

- Click **Add**
- Enter a **Name** for the site
- Enter a **Description** for the site
- If needed, **Add a Network Region**
- **Enable** Location Based Routing
- Choose an **Emergency Calling Policy**
- Choose an **Emergency Call Routing Policy**
- Add one or more **Network Subnets**
- **Save**

## PowerShell

### Network Region Commands

- [New-CsTenantNetworkRegion](#)
- [Get-CsTenantNetworkRegion](#)
- [Remove-CsTenantNetworkRegion](#)
- [Set-CsTenantNetworkRegion](#)

PowerShell example to create a new Network Region

```
New-CsTenantNetworkRegion -NetworkRegionID "US"
```

### Network Site Commands

- [New-CsTenantNetworkSite](#)
- [Get-CsTenantNetworkSite](#)
- [Remove-CsTenantNetworkSite](#)
- [Set-CsTenantNetworkSite](#)

PowerShell example to create a new Network Site added to a Network Region

```
# Get a list of Network Regions
Get-CsTenantNetworkRegion

# Define the Network Site Properties
$NetworkSiteProperties = @{

    NetworkSiteID           = "Warehouse Office"
    NetworkRegionID         = "US"
    EnableLocationBasedRouting = $true
    EmergencyCallingPolicy   = "Warehouse Office"
    EmergencyCallRoutingPolicy = "EvolveIP-ECRP-US-West"
    Description              = "West Coast Warehouse"
}

New-CsTenantNetworkSite @NetworkSiteProperties
```

### Network Subnet Commands

- [New-CsTenantNetworkSubnet](#)
- [Get-CsTenantNetworkSubnet](#)
- [Remove-CsTenantNetworkSubnet](#)
- [Set-CsTenantNetworkSubnet](#)

PowerShell example to add an IPv4 Subnet to a Network Site.

```
# Define the Network Subnet Properties
$SubnetProperties = @{

    SubnetID       = "10.10.30.0"
    MaskBits       = "24"
    NetworkSiteID  = "Warehouse Office"
    Description     = "West Coast Warehouse Office"
}

# Add the network subnet to the network site
New-CsTenantNetworkSubnet @SubnetProperties
```

PowerShell example to add an IPv6 Subnet to a Network Site

```
# Define the Network Subnet Properties
$SubnetProperties = @{

    SubnetID       = "2001:4898:e8:25:844e:926f:85ad:dd8e"
    MaskBits       = "120"
    NetworkSiteID  = "Warehouse Office"
    Description     = "West Coast Warehouse Office"
}

# Add the network subnet to the network site
New-CsTenantNetworkSubnet @SubnetProperties
```

PowerShell example to bulk add Subnets to Network Sites, which requires a CSV file with the following Headers (Column Names):

- SubnetID
- MaskBits
- NetworkSiteID
- Description

```
# Import the CSV file into a variable
$Subnets = Import-Csv -Path "C:\Path\to\Csvfile.csv"

ForEach ($Subnet in $Subnets) {

    # Define the Network Subnet Properties
    $SubnetProperties = @{

        SubnetID      = "$($Subnet.SubnetID)"
        MaskBits      = $Subnet.MaskBits
        NetworkSiteID = "$($Subnet.NetworkSiteID)"
        Description    = "$($Subnet.Description)"
    }

    # Add the network subnet to the network site
    New-CsTenantNetworkSubnet @SubnetProperties
}
```