Bit9 + Cb CARBON BLACK	Carbon Black Cluster Deployment Guide 12-Jun-2014 Cb-Support@Bit9.com

Introduction

The purpose of this document is to define basic configuration guide for a multi-clustered environment in which a Carbon Black cluster contains multiple master nodes and any number of associated minion nodes. The goal is to give a basic understanding of inter-nodal communications, design considerations, sizing criteria and server placement.

Sections

Cluster Communications

The basics of cluster communications consist of sensors checking in to the master node and pushing data to the assigned minion node. Node assignment is based on the sensor ID and determined initial check-in. The master node will equally distribute new sensors across the minion nodes.

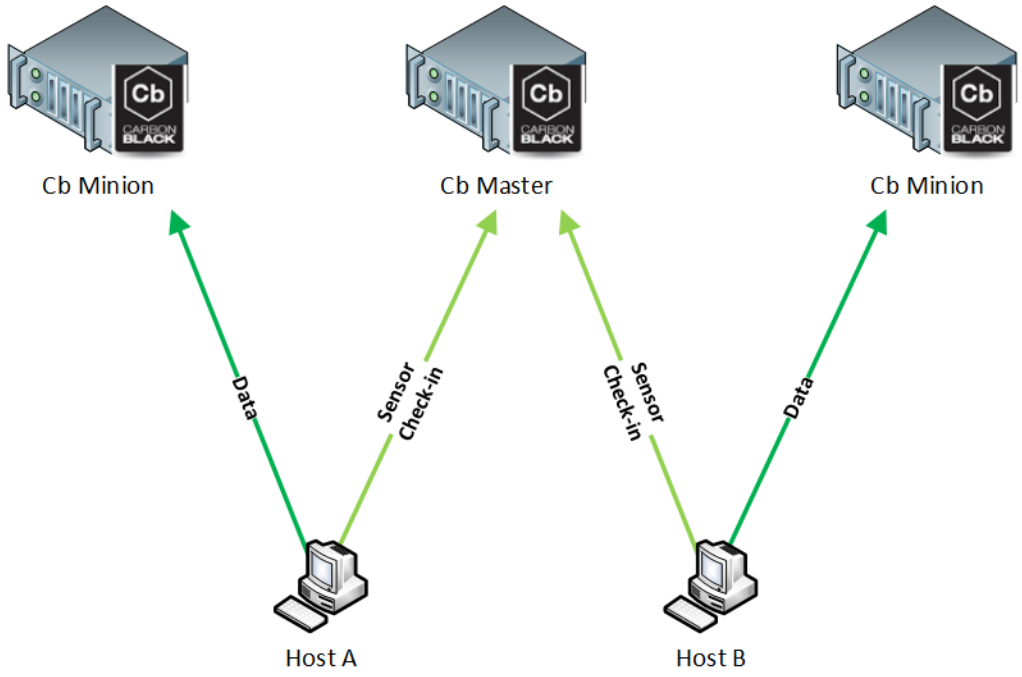

Figure 1 Cluster Communication

Figure 1 is a graphical representation of sensor to cluster communications. All sensor communication occurs over port 443 unless the sensor port is changed in the */etc/cb/nginx/conf.d/cb.conf* file. An example configuration can be found at */etc/cb/nginx/conf.d/cb-multihome.conf*.

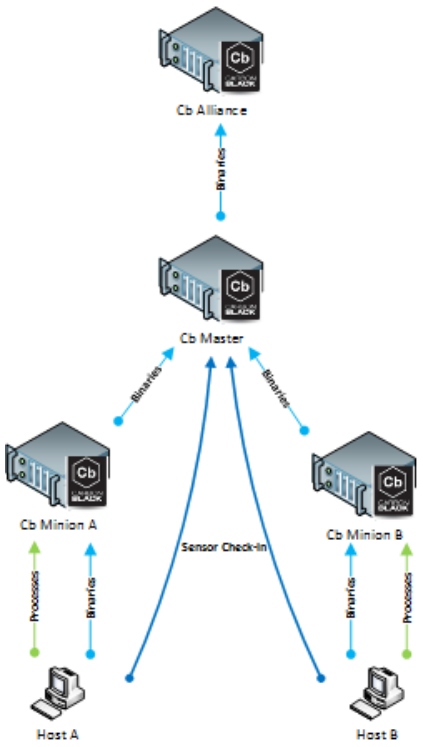

Figure 2 Cluster Data Flow

Figure 2 displays the actual sensor to cluster data flow. Note that with the exception of sensor check-in activity all data flow to or through the appropriate minion node. If Alliance participation is selected, the master node will receive a copy of all binaries for transmittal to the Alliance server for analysis.

Best Practice: Do not configure any shards on the master node as a general rule of thumb. The master should be utilized to coordinate and facilitate control of all sensors and as a focal point for Alliance communications. The minion nodes perform the difficult task of SOLR storage, indexing and retrieval.

Hardware Requirements

Refer to the *Carbon Black – Enterprise Server Sizing Guide* for a clustered installation's hardware requirements.

Node Placement

Master nodes and minion nodes should be placed in the same rack on the same network segment with a high-speed connection between nodes.

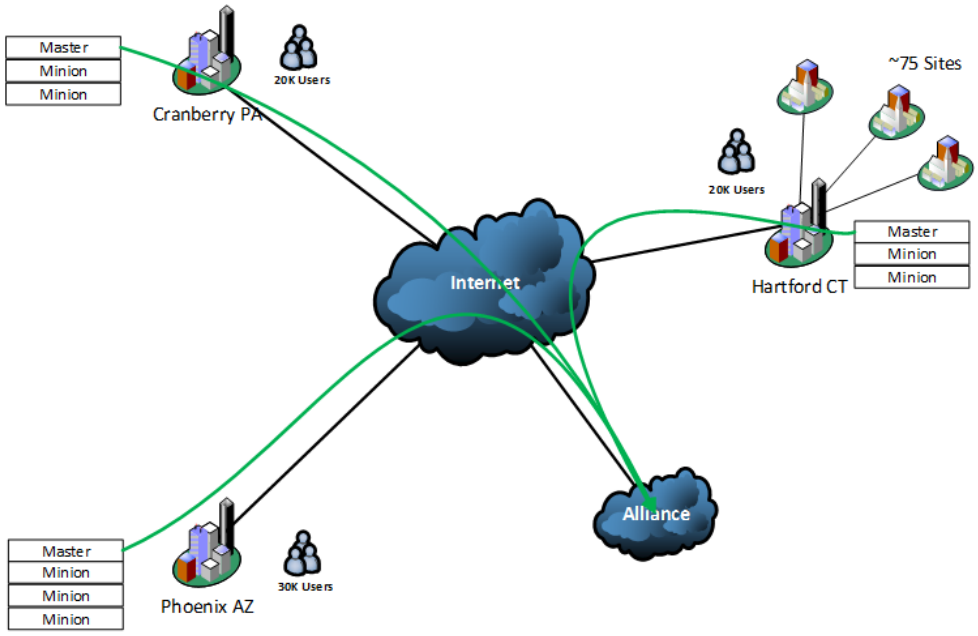

Figure 3 Multiple Cluster Configuration

When multiple data centers are involved placement of master and minion nodes should be distributed per Figure 3. A master node will be placed at each data center with at least 1 minion per 5K hosts. The SOLR data store should be distributed across the nodes with at least 2 shards per minion. This configuration achieves load balancing and provides the capacity to expand minion nodes in the future without re-sharding the entire data store. Using the Phoenix site as an example we would setup 6 shards when running *cbinit* with the *--proc-store-shards=6* when the master node is initialized.

Bandwidth Requirements

Bandwidth is the biggest factor to consider for remote sites. A default Windows 7 install with a normal office user will generate an average 200 bps to 1.2 kbps in 4.0 and earlier releases, 4.1 (mid-Feb release) adds compression and will reduce requirements to 50-200 bps. On low bandwidth remote links it might be necessary to put a master node at the site and have sensors register to the master node.

DMZ Configuration

For remote users a master node can have multiple interfaces to allow the configuration of a public facing interface for sensor communication as depicted in Figure 4.

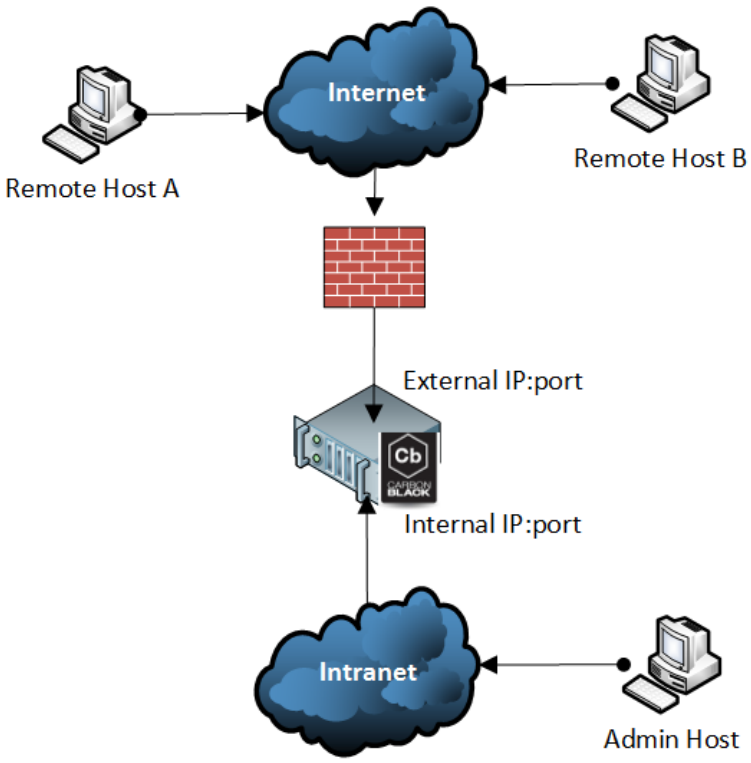

Figure 4 DMZ Configuration

An example configuration can be found in */etc/cb/nginx/conf.d/cb-multihome.conf.example* and includes instruction at the top of the file for configuration.

This file breaks up the configuration into two server {} directives. The first section is for the **SENSORs:**

```
server
{
  # This server configuration is used for communications between the sensors
    # and the server.

    listen [::]:80 ipv6only=off;
    listen [::]:443 ssl ipv6only=off;
```

The second section is for the **UI** and the **API**:

```
server
{
    # This server configuration is used for CB Enterprise Server's Web UI

    listen [::]:80 ipv6only=off;
    listen [::]:443 ssl ipv6only=off;
```

Individual IP addresses or port bindings can be specified via the listen directive for either/both. More documentation for the listen directive:

http://nginx.org/en/docs/http/ngx_http_core_module.html#listen

It is important to note that if you choose to change the sensor port in the first section of this file, you will also need to change it on clients already deployed and the port setting in the UI used for .msi and .exe creation.

Recommended Deployment Steps

1. Provision and build master nodes at each site following the instruction in the "HOWTO-Carbon Black Cluster Configuration" guide. During the master node *cbinit* process utilize the *--proc-store-shards=PROC_STORE_SHARDS* where the PROC_STORE_SHARDS is equal to twice the number of minion nodes and minion nodes equal preferred sensor density per minion. The maximum density recommended per minion is 10K sensors on a well provisioned server.

2. From the master node generate the preferred (.exe or .msi) deployment package for your environment. It is recommended not begin deployment until minion nodes are operational. While shards can be moved after they contain records, a good best practice is to set up the Carbon Black cluster first and then add sensors.

3. Standup minion nodes and utilize the */usr/share/cb/cbcluster add-node* command from the master to join the minion node to the cluster. During this process you will be asked which shards will be managed by this node. Move the appropriate shards to the minion node. The minion node process is detailed in "HOWTO-Carbon Black Cluster Configuration".

4. From the master node at each site generate the preferred deployment package and begin deploying sensors.