V-NetLab Host Manager Interface Document

V-NetLab Host Manager accepts requests from Group Manager, starts up (or shuts down) related vmnets, and calls VMWare API to start (or shutdown/suspend/resume) the virtual machines belonging to a virtual network. Also, it propagates the "routing information" generated by Group Manager to Kernel Module so that data link layer virtualization will be enforced properly. So there are two interfaces for Host Manager: the GrpMgr-HostMgr Interface and the HostMgr-Kernel Module Interface.

1. GrpMgr-HostMgr Interface

There are two levels of interaction between Group Manger and Host Manager: sockets and NFS. Sockets are used for sending requests and results[1]. NFS is used for sharing files. For instance, via sockets Group Manager sends a start request to Host Manager, on receiving the request, via NFS Host Manager can directly access the network configuration files generated by the virtualization engine of Group Manager, starts the network and then sends back the results to Group Manager through sockets. The following section discusses these two levels of interaction in more detail.

1.1 Sockets Communication Between Group Manager and Host Manager

1.1.1 Message Format

As shown below, the message between Group Manager and Host Manger has two parts: the header and the data.

```
typedef struct __GMGR_HMGR_MSG__ {
    gmgr_hmgr_hdr header;
    char data[GrpMgr_HostMgr::MAX_DATA_LEN];
} gmgr_hmgr_msg;
```

The data field is used for sending additional information in certain types of messages (such as ssh port numbers for VMs). The structure for header is as follows:

```
typedef struct __GMGR_HMGR_HEADER__ {
    int opid;      // request type or result
    int uid;       // user id
```

[1] Besides sending requests and results, sockets are also used for checking whether Group Manager is alive when Host Manager just starts up. Since this is a minor issue, we omit it for simplicity.

```
    int nid;       // network id
    int tid;       // team id
    int team;      // team network or not
    int console ; // request from console or not
} gmgr_hmgr_hdr ;
```

Possible values of certain fields of the header is shown in the following table.

opid	GrpMgr_HostMgr::START, GrpMgr_HostMgr::SHUTDOWN, GrpMgr_HostMgr::SUSPEND2, GrpMgr_HostMgr::RESUME2, GrpMgr_HostMgr::DEREGISTER, GrpMgr_HostMgr::SUCCESS, GrpMgr_HostMgr::FAILURE
team	TEAM_YES, TEAM_NO
console	CONSOLE_YES, CONSOLE_NO

1.1.2 Protocol

To start/shutdown/suspend/resume a network, user uses VNetMgr (i.e. the user interface) to forward the requests to GrpMgr, after allocating resources and generating the configuration files for the network, GrpMgr notifies involved HostMgr to finish the task. After the job is done, Host Manager sends back the result to GrpMgr, which will forward it to VNetMgr.

This section discusses the protocol between Group Manger and Host Manager. To make the whole procedure clear and complete, we also include communications between VNetMgr and GrpMgr here. Following are the protocols for starting a network, shutting down a network, suspending a network and resuming a network.

- Starting a network

Step	VNetMgr	Msg Type & Direction	GrpMgr	Msg Type & Direction	HostMgr
1		OP_MGR_SD_N→			
2				GrpMgr_HostMgr::SHUTDOWN→	
3				←GrpMgr_HostMgr::SUCCESS	
4		←START_SUCCESS			

- Shutting down a network

Step	VNetMgr	Msg Type & Direction	GrpMgr	Msg Type & Direction	HostMgr
1		OP_MGR_SD_N→			
2				GrpMgr_HostMgr::SHUTDOWN→	
3				←GrpMgr_HostMgr::SUCCESS	
4		←SHUTDOWN_SUCCESS			
5		OP_MGR_DEREG_NW→			
6				GrpMgr_HostMgr::DEREGISTER→	
7				←GrpMgr_HostMgr::SUCCESS	

8		←_SUCCESS			

- Suspending a network (suspend2)

Step	VNetMgr	Msg Type & Direction	GrpMgr	Msg Type & Direction	HostMgr
1		OP_MGR_SU2_N→			
2				GrpMgr_HostMgr::SUSPEND2→	
3				←GrpMgr_HostMgr::SUCCESS	
4		←SUSPEND_SUCCESS			
5		OP_MGR_DEREG_NW→			
6				GrpMgr_HostMgr::DEREGISTER→	
7				←GrpMgr_HostMgr::SUCCESS	
8		←_SUCCESS			

- Resuming a network (resume2)

Step	VNetMgr		GrpMgr		HostMgr
1		OP_MGR_RE2_N→			
2				GrpMgr_HostMgr::RESUME2→	
3				←GrpMgr_HostMgr::SUCCESS	
4		←RESUME_SUCCESS			

1.2 Files shared via NFS

On receiving a request to start or resume2 a virtual network, GrpMgr first generates all configuration files and places them under NFS directory /home/vnetlab/managed/userNetworks (or /home/vnetlab/managed/teamNetworks if it's a team network). HostMgr can access them afterwards to finish the job required.

For each user, a directory with user id will be created. For example, if user 1008 has registered a network net5, then following directory will be created: /home/managed/userNetworks/1008/net5.

The following section describes the configuration files in this directory as well as their formats.

.netConfigInfo: config file for the virtual machines as specified by user

VmMacAddr	VmName	VmIp	BcastAddr	Subnet	Netmask

.uid : The user id of the user who last started the network

uid

vmSrc.map: Specifies which virtual machine is on which host

VmName	VmSrcDir	PhyHost

Suppose net5 has virtual machines on hosts v3 and v4, then we will also have the following directories in /home/managed/userNetworks/1008/net5.

v3: Directory containing configuration information for all VMs on host v3
v4: Directory containing configuration information for all VMs on host v4

Within directory of v3 or v4, following files are maintained.

vm1.cfg: The vmware configuration file for virtual machine vm1
vm2.cfg: The vmware configuration file for virtual machine vm2

port2vm.map : Specifies port forwarding information from physical to virtual machine.

ssh port no.	VmMacAddr	VmIp	VInfId	VInfMacAddr	VInfIp

pt.map: The participation table (sibling closure) for this virtual network

src sibling closure id	dest sibling closure id	no. of local vmnetX interface	list of local vmnetX interface	no. of remote mac addrs	list of remote mac addrs

vinf.map: Virtual interface configuration information used by Vnetlab

VInfId	VInfId	sibling closure id	VInfIp	Netmask	dhcpdX.conf

(VInf = virtual interface, it is actually vmnetX)

vm2hub.map: Specifies which VM is connected to which virtual interface

VmMacAddr	VInfId	sibling closure id	on_this_phy_host

vm.loc: Specifies the actual location of the virtual machine on physical host

VmName	VmWorkspaceDir

2. HostMgr-Kernel Module Interface

Please refer to "V-NetLab Kernel Module Interface Document" written by Manish.