Chapter 18

NIC Configuration Scenarios

This chapter provides detailed descriptions of the NIC configuration scenarios included in the sample data. This chapter contains the following sections:

Overview of NIC Configuration Scenarios

The NIC configuration scenarios in the sample data provide resolutions for a variety of network configurations.

Each NIC scenario includes two types of configuration:

- Centralized—A single host configuration for use with NIC replication. In a centralized configuration all agents and resolvers reside on one host. The name of this host is DemoHost.
- Distributed—A multiple host configuration in which agents and resolvers are distributed among more than one host. This type of configuration is designed for use with NIC host redundancy. In most cases, the hosts are named OnePopH1 (a host in a pop) and OnePopBO (a host in a back office).

The best way to view the sample data is with the NIC Web Admin tool (see *Chapter 15, Managing the NIC*).

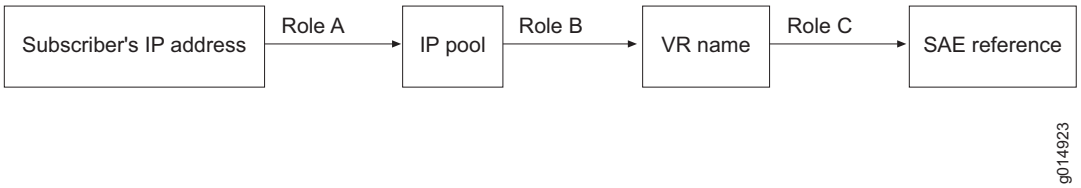
For a summary of the NIC configuration scenarios included in the sample data, see *Chapter 12, Locating Subscriber Information*.

OnePop Scenario

The OnePop scenario illustrates a configuration that supports one POP and provides no redundancy. The realm for this configuration accommodates the situation in which IP pools are configured locally on each VR. The resolution process takes a subscriber's IP address as the key and returns a reference to the SAE managing this subscriber as the value.

Figure 23 shows the resolution graph for this realm.

Figure 23: Resolution Process for IP Realm



The following agents collect information for resolvers in this realm:

- Directory agent PoolVr collects and publishes information about the mappings of IP pools to VRs.
- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs.

The OnePop sample provides two host configurations: a centralized configuration for use with NIC replication and a distributed configuration that uses NIC redundancy.

Centralized Configuration

In this configuration, single host DemoHost supports all agents and resolvers. When the NIC proxy sends a subscriber's IP address to host DemoHost, the following sequence of actions occurs:

1. The host passes the IP address to resolver A1.
2. Resolver A1 obtains an IP pool for the IP address and forwards the request to resolver B1.
3. Resolver B1 obtains a VR name for the IP pool and returns the VR name to resolver A1.
4. Resolver A1 forwards the VR name to resolver C1.

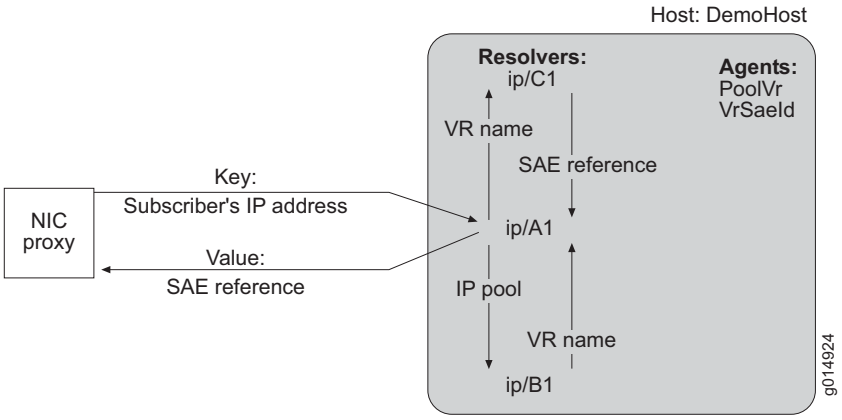
5. Resolver C1 obtains an SAE reference for the VR and returns the VR identity to resolver A1.

6. Resolver A1 passes the SAE reference to its host.

7. The host returns the SAE reference to the NIC proxy.

Figure 24 shows the interactions of the NIC components for this realm.

Figure 24: OnePop Centralized Configuration

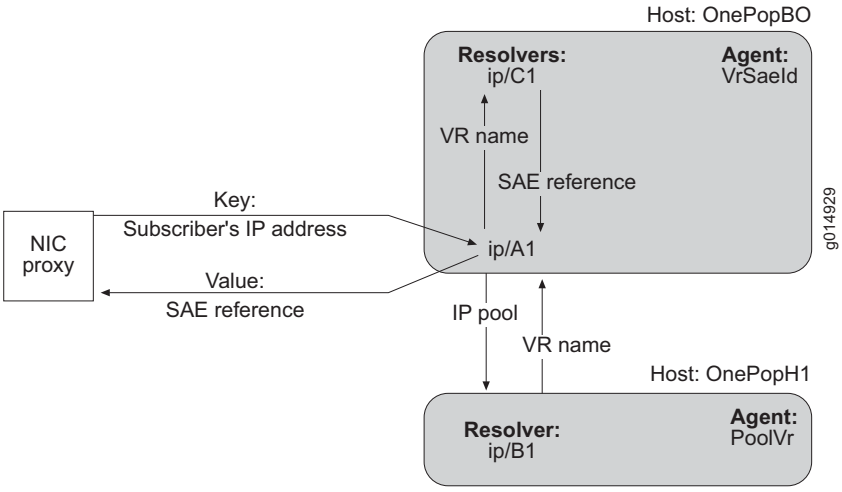


Distributed Configuration

In this configuration, the agents and resolvers are distributed among several hosts. When the NIC proxy sends a subscriber's IP address to host OnePopBO, the components execute the same actions as they do in the centralized configuration (see *Centralized Configuration* on page 320).

Figure 25 illustrates the interactions of the NIC components for this realm.

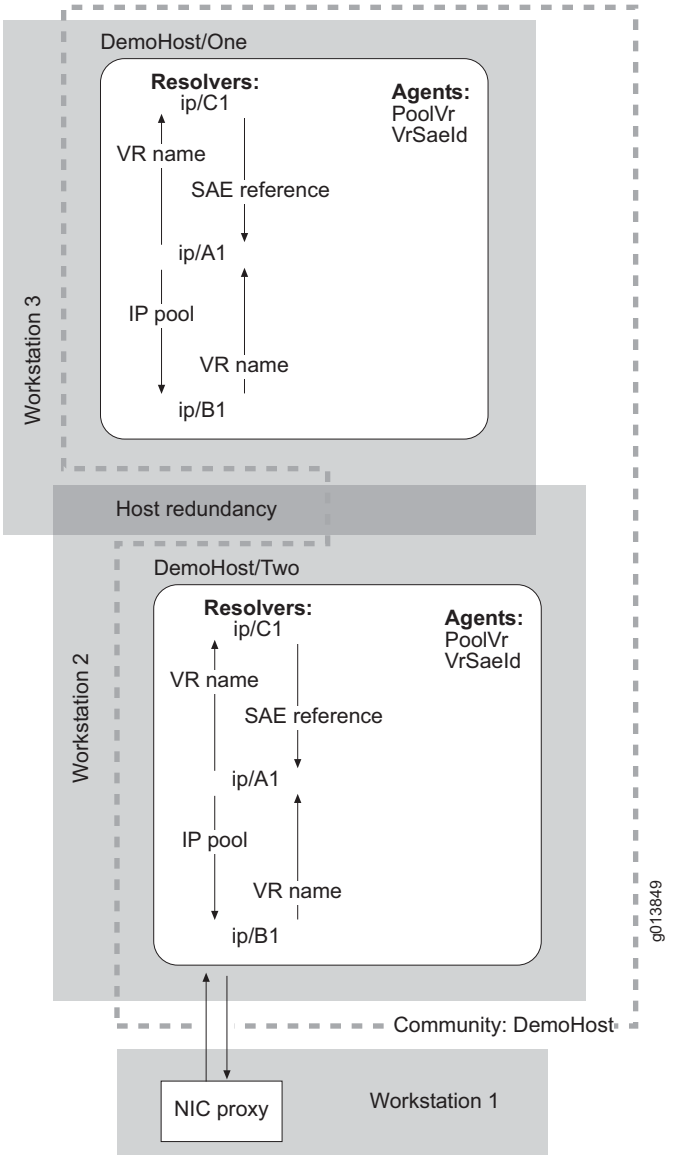
Figure 25: OnePop Distributed Configuration



Redundancy

This sample data includes host redundancy for the centralized configuration. The hosts DemoHost/One and DemoHost/Two, which are installed on different machines, provide host redundancy. These hosts form the community DemoHost, which does not include a monitor.

Figure 26: Redundancy for OnePop Centralized Configuration



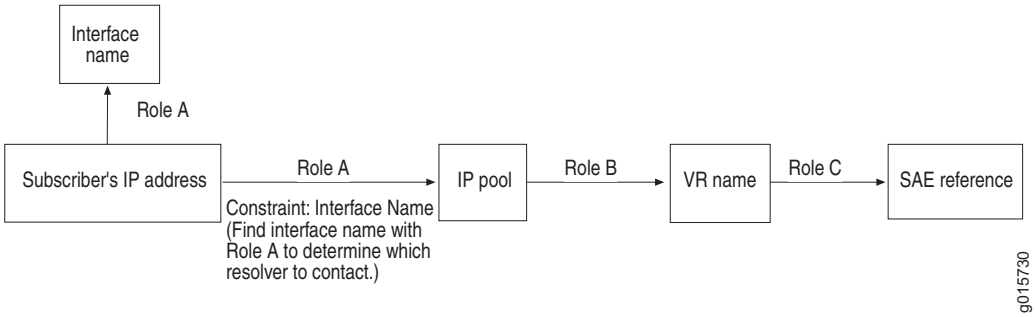
OnePopDynamicIp Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario and provides no redundancy. The realm for this configuration accommodates the situation in which IP pools are configured locally on each virtual router object. The resolution process takes a subscriber's IP address as the key and returns a reference to the SAE managing this subscriber as the value.

The scenario supports a configuration scenario for a PacketCable Multimedia Specification (PCMM) environment in which you use the assigned IP subscriber method to log in subscribers, and use the NIC to determine the subscriber's SAE.

Figure 27 shows the resolution graph for this realm.

Figure 27: Resolution Process for IP Realm



The following agents collect information for resolvers in this realm:

- Directory agent PoolVr collects and publishes information about the mappings of IP pools to VRs.
- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs.

The OnePopDynamicIp scenario provides two host configurations: a centralized configuration and a distributed configuration.

Centralized Configuration

In this configuration, single host DemoHost supports all agents and resolvers. When the NIC proxy sends a subscriber's IP address to host DemoHost, the following sequence of actions occurs:

1. The host passes the IP address to resolver A1.
2. Resolver A1 obtains an IP pool name and interface name for the IP address and forwards the request to resolver B1.
3. Resolver B1 obtains a VR name for the IP pool name and interface name and returns the VR name to resolver A1.
4. Resolver A1 forwards the VR name to resolver C1.

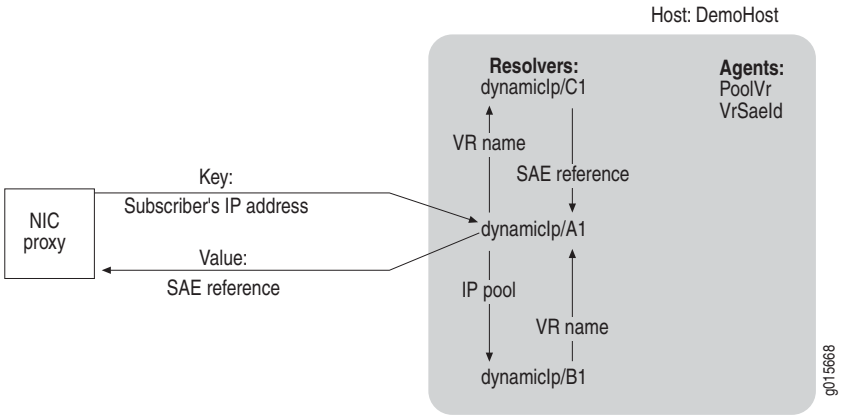
5. Resolver C1 obtains an SAE reference for the VR and returns the VR identity to resolver A1.

6. Resolver A1 passes the SAE reference to its host.

7. The host returns the SAE reference to the NIC proxy.

Figure 28 illustrates the interactions of the NIC components for this realm.

Figure 28: OnePopDynamicIp Centralized Configuration

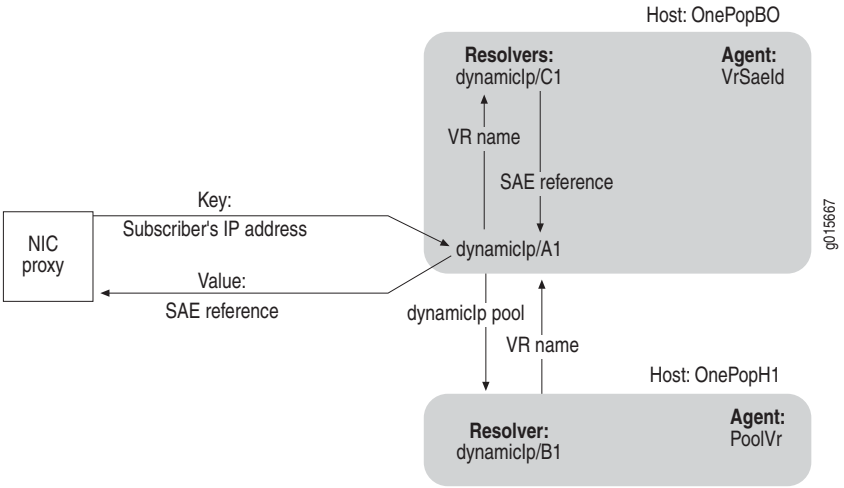


Distributed Configuration

In this configuration, the agents and resolvers are distributed among several hosts. When the NIC proxy sends a subscriber's IP address to host OnePopBO, the components execute the same actions as they do in the centralized configuration (see *Centralized Configuration* on page 323).

Figure 29 illustrates the interactions of the NIC components for this realm.

Figure 29: OnePopDynamicIp Distributed Configuration

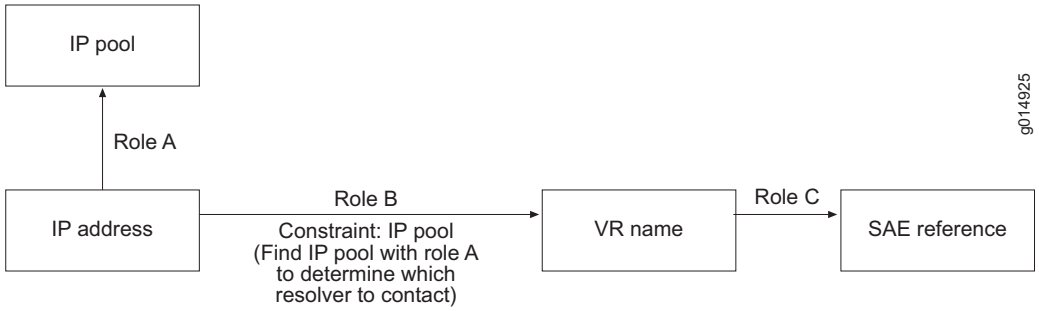


OnePopSharedIp Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario and provides no redundancy. However, the realm for this configuration accommodates the situation in which IP pools are shared by VRs in the same POP. The resolution process takes a subscriber's IP address as the key and returns a reference to the SAE managing this subscriber as the value.

Figure 30 shows the resolution graph for this realm.

Figure 30: Resolution Process for SharedIp Realm



The following agents interact with resolvers in this realm:

- SAE plug-in agent IpVr collects and publishes information about the mappings of IP addresses to VRs.
- Directory agent PoolVr collects and publishes information about the IP pools used by the VRs in a POP. Because the IP pools are shared between VRs, this agent discards information about VRs.
- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs.

The OnePopSharedIP scenario provides two host configurations: a centralized configuration and a distributed configuration.

Centralized Configuration

In this configuration, single host DemoHost supports all agents and resolvers. When the NIC proxy sends a subscriber's IP address to host DemoHost, the following sequence of events occurs:

1. The host passes the IP address to resolver A1.
2. Resolver A1 obtains an IP pool for the IP address.
3. Resolver A1 forwards the IP address and the IP pool to resolver B1.
4. Resolver B1 obtains a VR name for the IP address and returns the VR name to resolver A1.
5. Resolver A1 forwards the VR name to resolver C1.

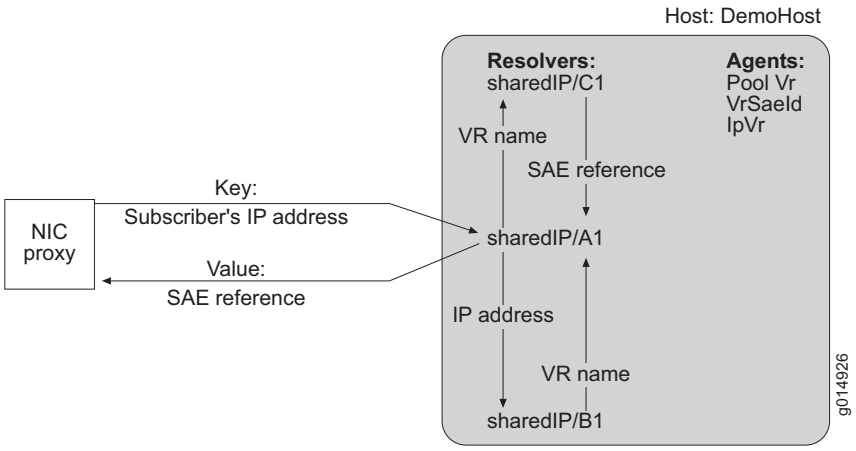
6. Resolver C1 obtains an SAE reference for the VR and returns the SAE reference to resolver A1.

7. Resolver A1 passes the SAE reference to its host.

8. The host returns the SAE reference to the NIC proxy.

Figure 31 shows the interactions of the NIC components for this realm.

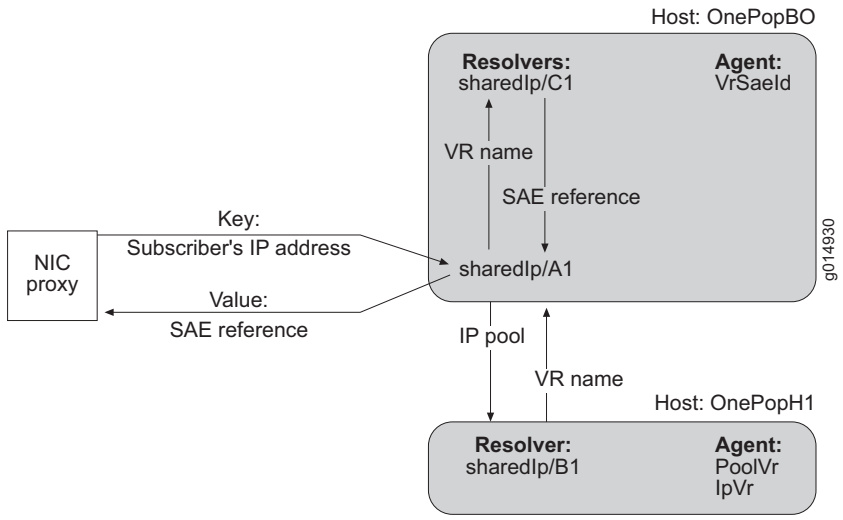
Figure 31: OnePopSharedIP Centralized Configuration



Distributed Configuration

In this configuration, the agents and resolvers are distributed among several hosts. When the NIC proxy sends a subscriber's IP address to the host OnePopBO, the resolvers execute the same actions as they do in the centralized configuration. Figure 32 illustrates the interactions of the NIC components for this realm.

Figure 32: OnePopSharedIP Distributed Configuration

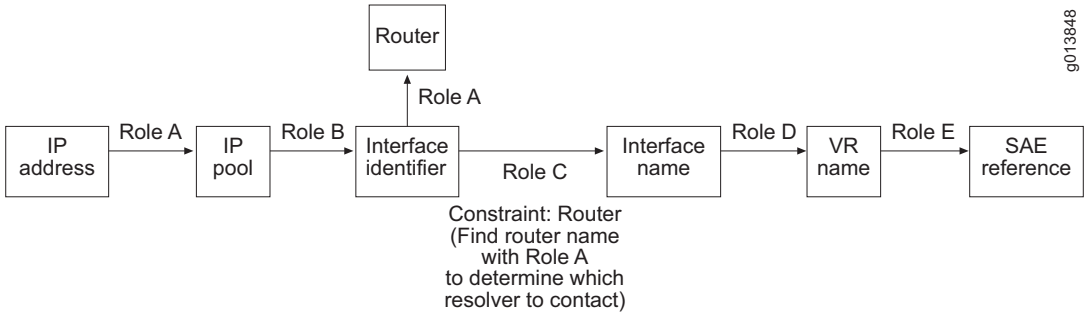


OnePopAssignedIp Scenario

This scenario illustrates a configuration that is similar to the OnePopShared scenario and provides no redundancy. However, the realm for this configuration accommodates situations in which the SAE manages the interface shared by many subscribers but does not directly manage the subscribers. The resolution process takes the subscriber's IP address as the key and returns the reference of the SAE managing this subscriber as the value.

Figure 33 shows the resolution graph for this realm.

Figure 33: Resolution Process for AssignedIp Realm



The following agents interact with resolvers in this realm:

- Router access agent PoolInterfaceId publishes information about the mappings of IP pools to interface identifiers from the router.
- Consolidator agent Router publishes information about the routers in the POP.
- SAE plug-in agent InterfaceIdInterface collects information about the mappings of interface identifiers to interface names and responds to resolution requests for this information.
- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs. Because this agent publishes information about all the currently configured VRs, it also provides information about the names of interfaces that a VR manages.

The OnePopAssignedIP scenario provides two host configurations: a centralized configuration and a distributed configuration.

Centralized Configuration

In this configuration, single host DemoHost supports all agents and resolvers. When the NIC proxy sends a subscriber's IP address to host DemoHost, the following sequence of events occurs:

1. The host passes the subscriber's IP address to resolver A1.

2. Resolver A1 determines the IP pool to which the subscriber's IP address belongs, and passes the IP address to resolver B1.

   Resolver A1 must find the most specific IP pool for the IP address. For example, an IP address of 192.9.4.10 belongs to both IP pools 192.9.0.0/16 and 192.9.4.0/24. The latter pool, however, is the more specific pool for this address.

   Consequently, resolver A1 needs to know about all configured IP pools. If resolver A1 and resolver B1 are on the same host, these resolvers receive the same events, and resolver A1 determines the IP pool information without using resolver B1. If, however, resolvers A1 and B1 are on separate hosts, resolver B1 provides the IP pool information to resolver A1.

3. Resolver B1obtains the identifier for the interface associated with the IP pool and returns the interface name to resolver A1.

4. Resolver A1 determines the router name directly from the interface identifier and forwards the interface identifier to resolver C1.

   In an environment with multiple POPs, resolver A1 uses the router name to determine the POP to which it should forward the request (see *AssignedIP Realm in Multiple POPs* on page 330).

5. Resolver C1 obtains an interface name for the interface identifier and returns the interface name to resolver A1.

6. Resolver A1 forwards the interface name to resolver D1.

7. Resolver D1 obtains a VR name for the interface name and returns the VR name to resolver A1.

8. Resolver A1 forwards the VR name to resolver E1.

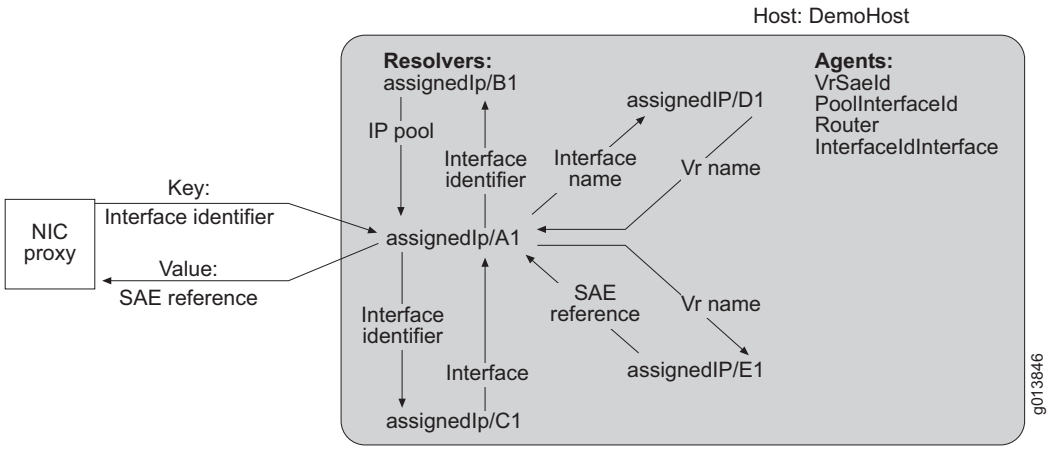
9. Resolver E1 obtains an SAE reference for the VR and returns the SAE reference to resolver A1.

10. Resolver A1 passes the SAE reference to its host.

11. The host returns the SAE reference to the NIC proxy.

Figure 34 shows the interactions of the NIC components for this realm.

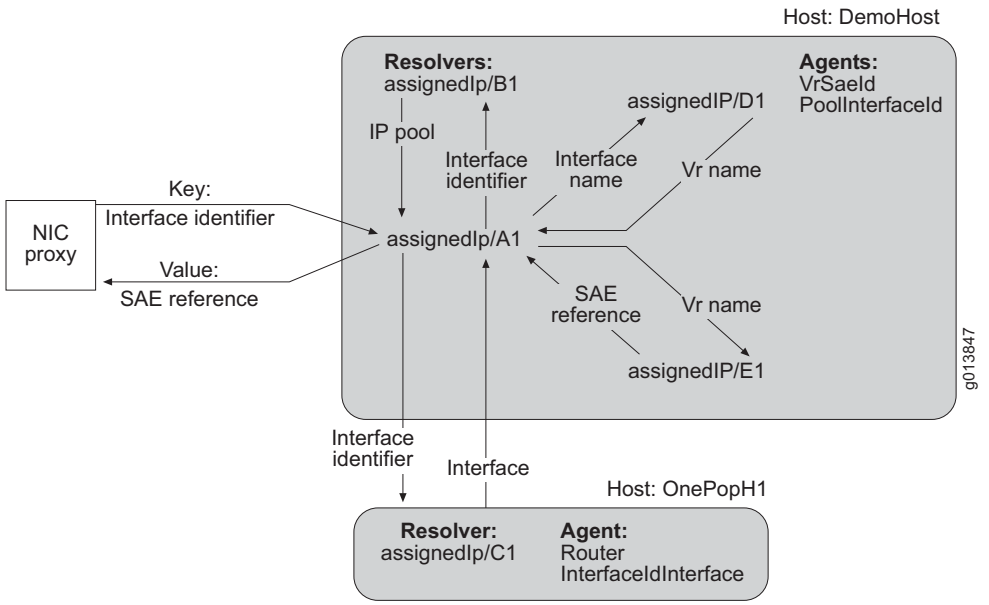
Figure 34: OnePopAssignedIP Centralized Configuration



Distributed Configuration

In this configuration, the agents and resolvers are distributed among several hosts. When the NIC proxy sends a subscriber's interface identifier to the host OnePopBO, the resolvers execute the same actions as they do in the centralized configuration. Figure 35 illustrates the interactions of the NIC components for this realm.

Figure 35: OnePopAssignedIP Distributed Configuration



AssignedIP Realm in Multiple POPs

To deploy the assignedIP realms in a multiple POP environment, deploy in each POP one set of role C resolvers and the agents InterfaceIdInterface and Router. You can deploy the other resolvers and agents in the back office.

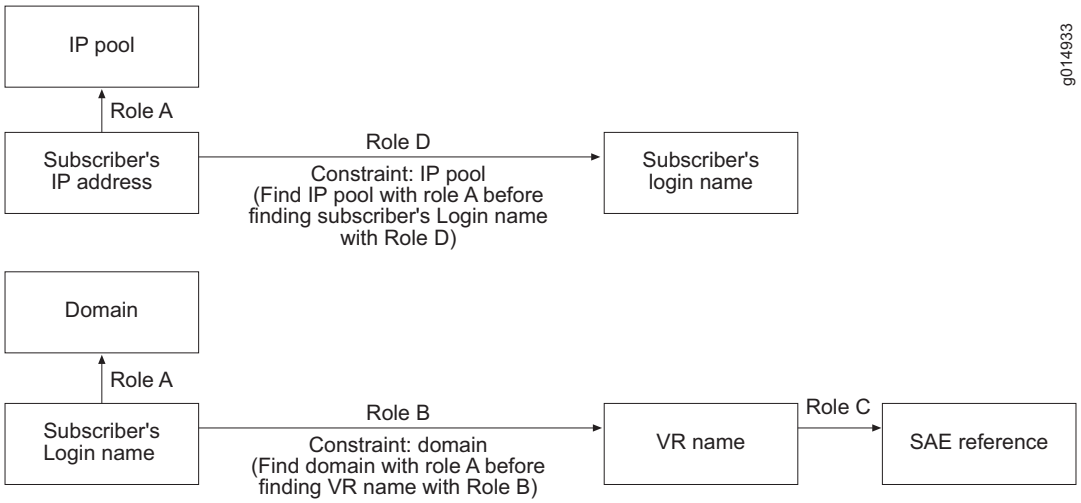
In a multiple POP environment, resolver A determines the POP to which it should send the request based on the router associated with the interface identifier.

OnePopLogin Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario and provides no redundancy. The realm for this configuration accommodates two independent resolution processes, which are used by volume-tracking applications (VTAs) and may be used for other purposes.

Figure 36 shows the resolution graphs for this realm.

Figure 36: Resolution Processes OnePopLogin Realm



The following agents interact with resolvers in this realm:

- SAE plug-in agent IpLoginName collects and publishes information about the mappings of IP addresses to login names.
- SAE plug-in agent LoginNameVr collects and publishes information about the mappings of login names to VRs.

- Directory agent Pool collects and publishes information about the IP pools used by the VRs in a POP. The agent uses the information about the IP pools to determine which resolver to communicate with, rather than communicating with all resolvers that are running role D.
- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs.

The OnePopLogin scenario provides two host configurations: a centralized configuration and a distributed configuration.

Centralized Configuration

In this configuration, single host DemoHost supports all agents and resolvers. Two NIC proxies are associated with this NIC configuration; one NIC proxy (called NIC proxy 1 in this documentation) submits subscribers' login names, and the other (called NIC proxy 2 in this documentation) submits subscribers' IP addresses.

When NIC proxy 1 sends a login name to the host DemoHost, the following sequence of events occurs:

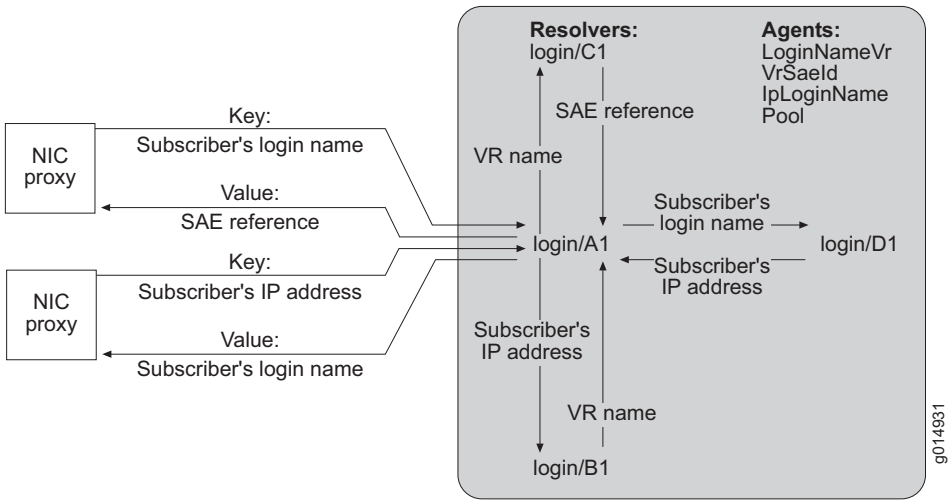
1. The host passes the login name to resolver A1.
2. Resolver A1 obtains a domain name for the login name.
3. Resolver A1 forwards the login name and the domain to resolver B1.
4. Resolver B1 obtains a VR name for the login name and returns the VR name to resolver A1.
5. Resolver A1 forwards the VR name to resolver C1.
6. Resolver C1 obtains an SAE reference for the VR and returns the SAE reference to resolver A1.
7. Resolver A1 returns the SAE reference to its host.
8. The host returns the SAE reference to the NIC proxy.

When NIC proxy 2 sends a subscriber's IP address to host DemoHost, the following sequence of events occurs.

1. The host passes the IP address to resolver A1.
2. Resolver A1 obtains an IP pool for the IP address.
3. Resolver A1 forwards the IP address and the IP pool to resolver D1.
4. Resolver D1 obtains a login name for the IP address and returns the login name to resolver A1.
5. Resolver A1 passes the login name to its host.
6. The host returns the login name to the NIC proxy.

Figure 37 illustrates the interactions of the NIC components for this realm.

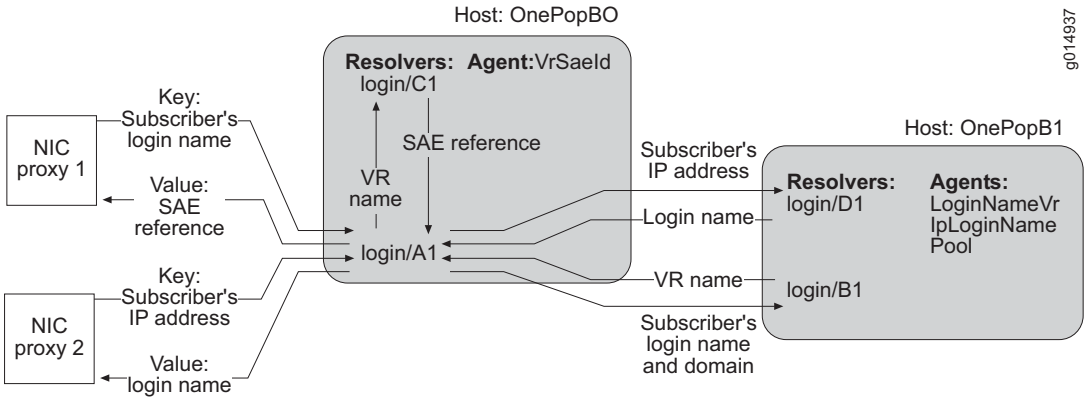
Figure 37: OnePopLogin Centralized Configuration



Distributed Configuration

In this configuration, the agents and resolvers are distributed among several hosts. When the NIC proxy sends a subscriber's IP address to the host OnePopBO, the resolvers execute the same actions as they do in the in the centralized configuration. Figure 38 illustrates the interactions of the NIC components for this realm.

Figure 38: OnePopLogin Distributed Configuration



OnePopDnSharedIp Scenario

The OnePopDnSharedIp scenario illustrates how to configure SAE plug-in agents that have state synchronization enabled to support an SAE plug-in that uses state synchronization. This scenario uses the same centralized and distributed configurations of hosts as the OnePop scenario.

Two realms are configured:

- Shared IP

  The resolution process is identical to that for the OnePopShared scenario (see Figure 30 on page 325).

- DN realm

  This realm uses essentially the same resolution process as the MultiPop DN realm (see Figure 46 on page 345). However, some of the constraints differ.

  This realm also uses the same agents as the MultiPop DN realm. The names of agents and resolvers are essentially the same as those in the MultiPop configuration, although they do not include a POP identifier. Figure 39 on page 334 illustrates the centralized configuration, and Figure 40 on page 336 illustrates the distributed configuration for the DN realms.

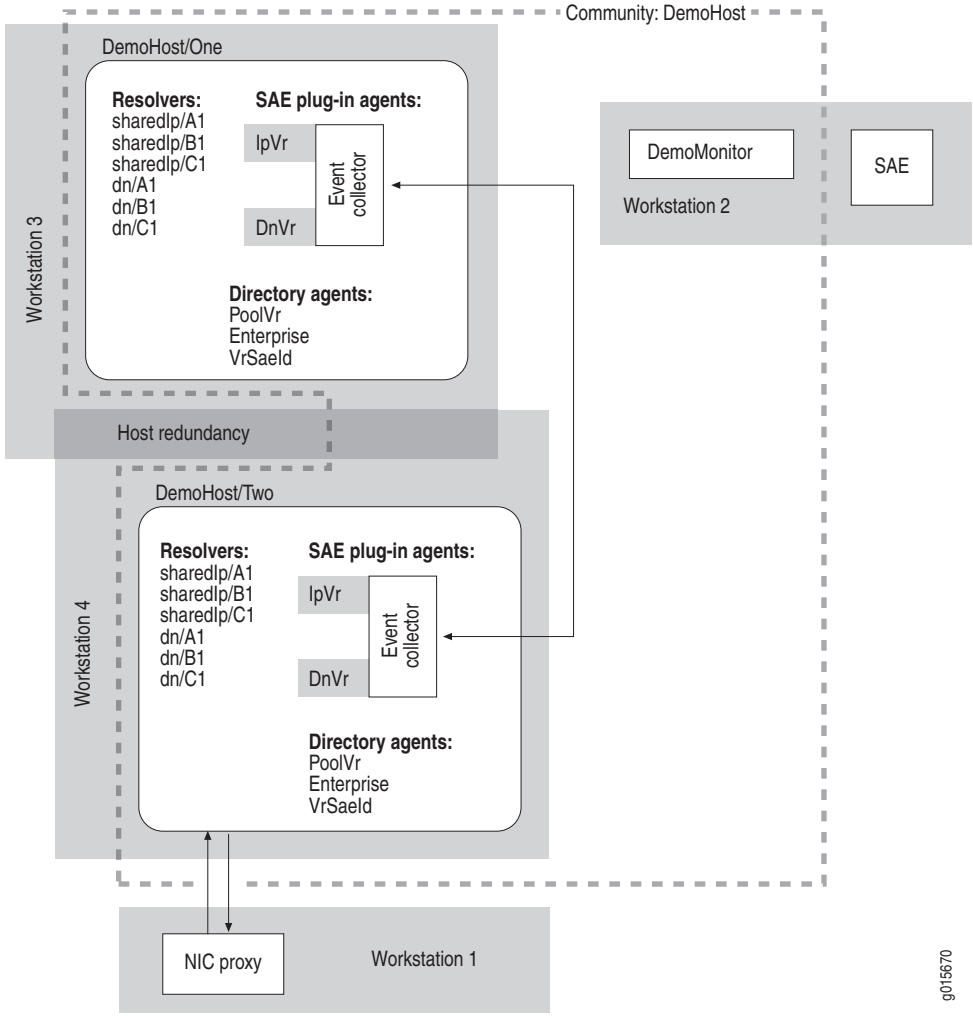
The configuration for the two realms is similar to the configuration for the shared IP and DN realms in the OnePopAllRealms scenario. See *OnePopAllRealms Scenario* on page 337. The OnePopAllRealms illustrates SAE plug-in agents configured to use SAE plug-in redundancy rather than SAE plug-in agents that have state synchronization enabled, which allows them to use NIC host redundancy.

Centralized Configuration

Figure 39 on page 334 shows the centralized configuration from the scenario. Host DemoHost supports all resolvers and agents. However, because host DemoHost is configured for redundancy, its redundant hosts (DemoHost/One and DemoHost/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host. The redundant hosts form a community called DemoHost with the monitor DemoMonitor, which tracks them.

The parent host DemoHost also supports two SAE plug-in agents, IpVr and DnVr, which share an event collector. Both plug-in agents have state synchronization enabled which means that the agents use HIC host redundancy.

Figure 39: OnePopDnSharedIp Realms Centralized Configuration



Distributed Configuration

Figure 40 on page 336 shows the distributed configuration from the scenario. Host OnePopBO supports two resolvers for each realm and a directory agent that is used by different realms. However, because host OnePopBO is configured for redundancy, its redundant hosts (OnePopBO/One and OnePopBO/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host.

Host OnePopH1 supports one resolver for each realm and agents that are used by different realms. Host OnePopH1 is also configured for redundancy, and its redundant hosts (OnePopH1/One and OnePopH1/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host.

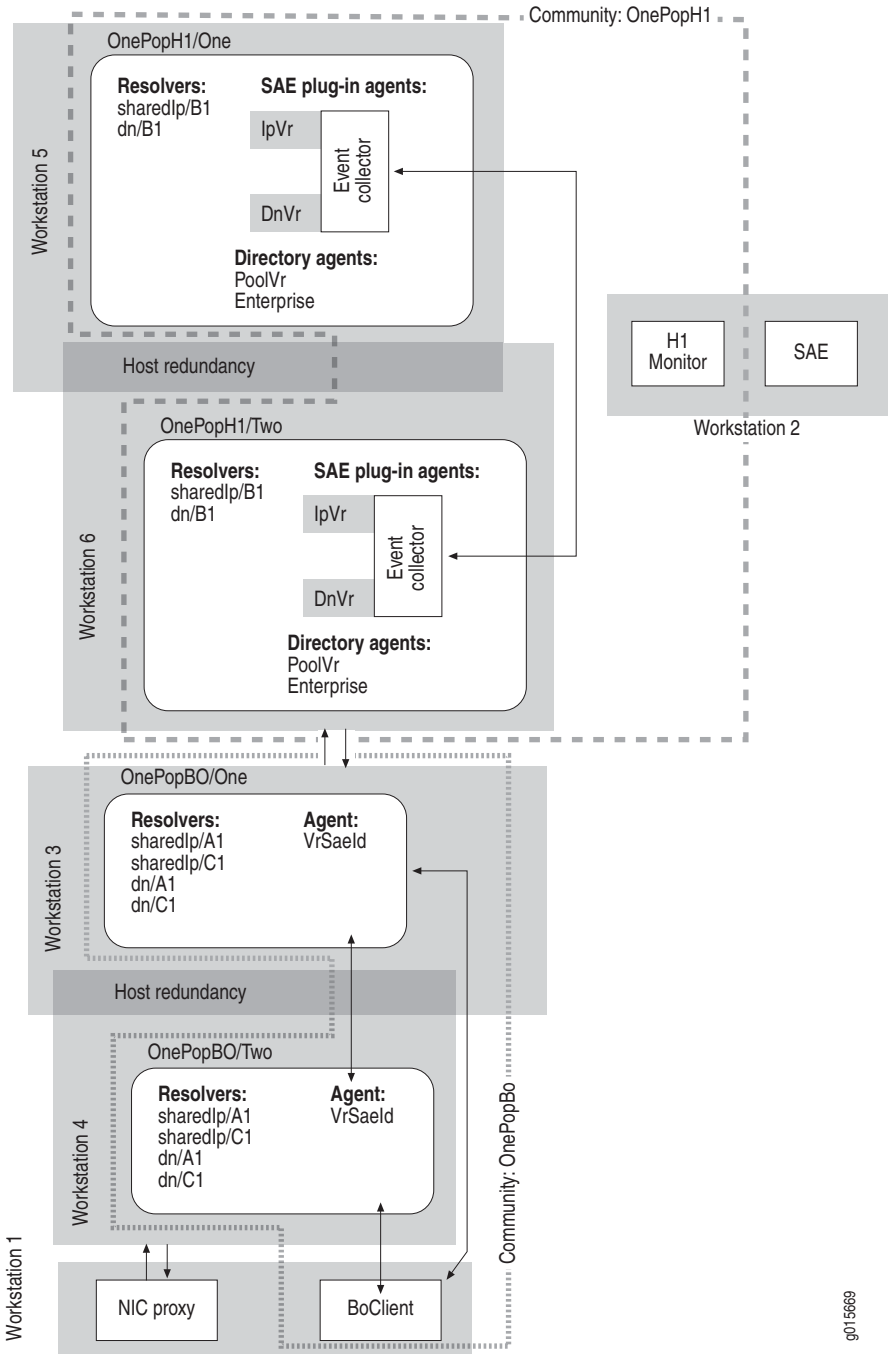
However, host OnePopH1 also supports two SAE plug-in agents, IpVr and DnVr, which share an event collector. These agents have state synchronization enabled and therefore use NIC host redundancy.

The redundant hosts OnePopBO/One and OnePopBO/Two are members of a community called OnePopBO. This community supports the monitor, BoClient, which is installed on the machine that supports the NIC proxy. BoClient tracks the connections between the redundant hosts OnePopBO/One and OnePopBO/Two from the point of view of the NIC client (NIC proxy).

Similarly, the redundant hosts OnePopH1/One and OnePopH1/Two are members of a community called OnePopH1. This community has one monitor, H1Monitor, which is located on the same machine as the SAE and tracks the connections among the redundant hosts in the same community, their primary host, and the other hosts in the configuration.

H1Monitor comprises the monitor process OnePop, which is installed on the same machine as the SAE. BoClient comprises the monitor process OnePopClient, which is installed on the same machine as the NIC proxy.

Figure 40: OnePopDnSharedIp Realms Distributed Configuration



OnePopAllRealms Scenario

The main purpose of the OnePopAllRealms scenario is to illustrate how to configure redundancy. This scenario uses the same centralized and distributed configurations of hosts as the OnePop scenario.

Three realms are configured:

- IP realm

  This realm uses essentially the same resolution process as the IP realm for the OnePop scenario (see Figure 23 on page 320). However, some of the constraints differ.

- Shared IP

  The resolution process is identical to that for the OnePopShared scenario (see Figure 30 on page 325).

- DN realm

  This realm uses essentially the same resolution process as the MultiPop DN realm (see Figure 46 on page 345). However, some of the constraints differ.

  This realm also uses the same agents as the MultiPop DN realm. The names of agents and resolvers are essentially the same as those in the MultiPop configuration, although they do not include a POP identifier. By reviewing the scenario, Figure 41 and Figure 42, you can determine exact pictures of the DN realms for the centralized and distributed configurations.

Centralized Configuration

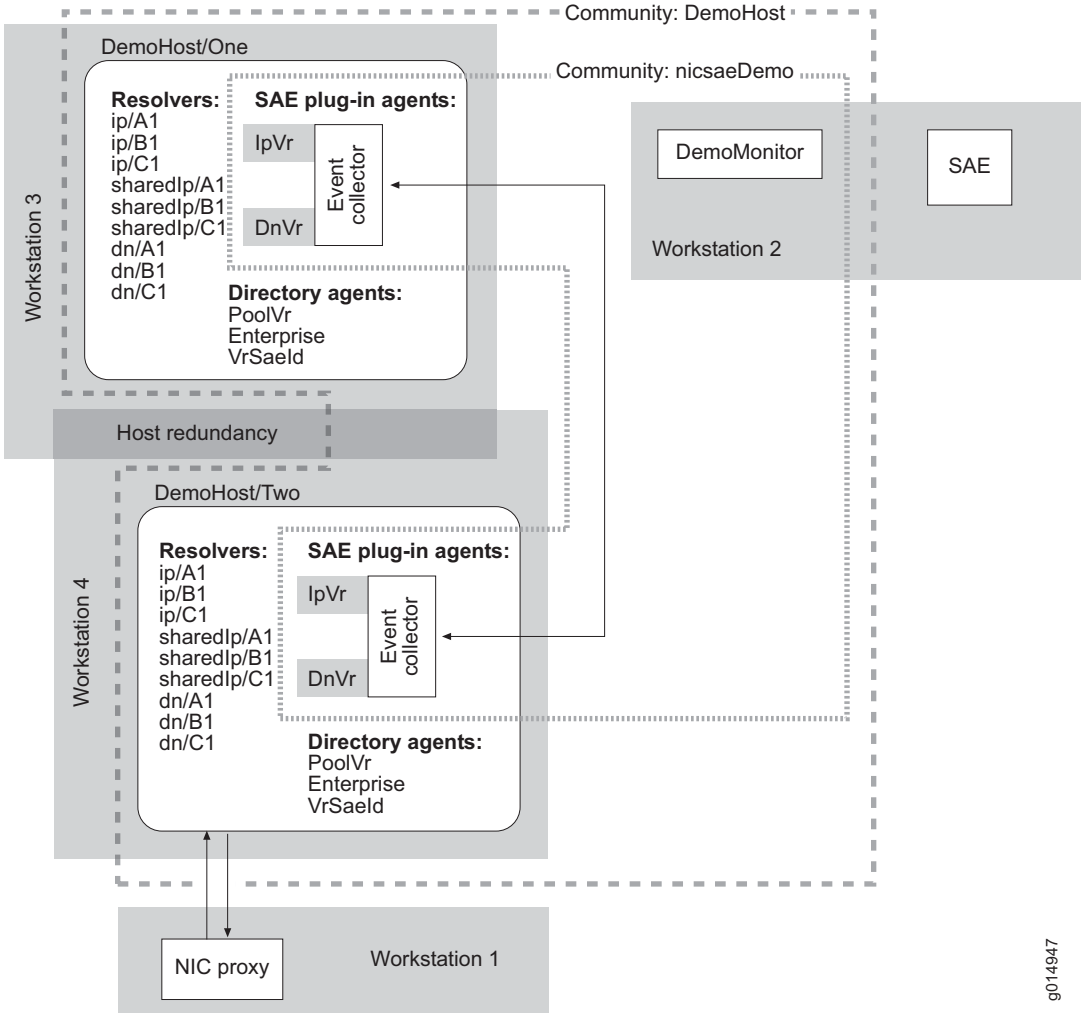
Figure 41 on page 338 shows the centralized configuration for the scenario. Host DemoHost supports all resolvers and agents. However, because host DemoHost is configured for redundancy, its redundant hosts (DemoHost/One and DemoHost/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host.

The parent host DemoHost also supports two SAE plug-in agents, IpVr and DnVr, which share an event collector. Each SAE plug-in agent has a redundant agent called Demo; these redundant agents also share an event collector. The redundant agents and their shared event collector are assigned to both redundant hosts DemoHost/One and DemoHost/Two.

The redundant agents form a community called nicsaeDemo with the monitor DemoMonitor, which tracks them. The redundant agents are identified in the community by the names DemoHost/One and DemoHost/Two; these names specify their hosts and provide unique identifiers for the redundant agents.

The redundant hosts form a community called DemoHost with the monitor DemoMonitor, which tracks them.

Figure 41: OnePopAllRealms Centralized Configuration



Distributed Configuration

Figure 42 on page 340 shows the distributed configuration for the scenario. Host OnePopBO supports two resolvers for each realm and a directory agent that is used by different realms. However, because host OnePopBO is configured for redundancy, its redundant hosts (OnePopBO/One and OnePopBO/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host.

Host OnePopH1 supports one resolver for each realm and agents that are used by different realms. Host OnePopH1 is also configured for redundancy, and its redundant hosts (OnePopH1/One and OnePopH1/Two) perform the host function. The redundant hosts are on different machines, and both hosts support the resolvers and agents assigned to the parent host.

However, host OnePopH1 also supports two SAE plug-in agents, IpVr and DnVr, which share an event collector. Each SAE plug-in agent has a redundant agent called onePop; these redundant agents also share an event collector. The redundant agents and their shared event collector are assigned to redundant hosts OnePopH1/One and OnePopH1/Two.

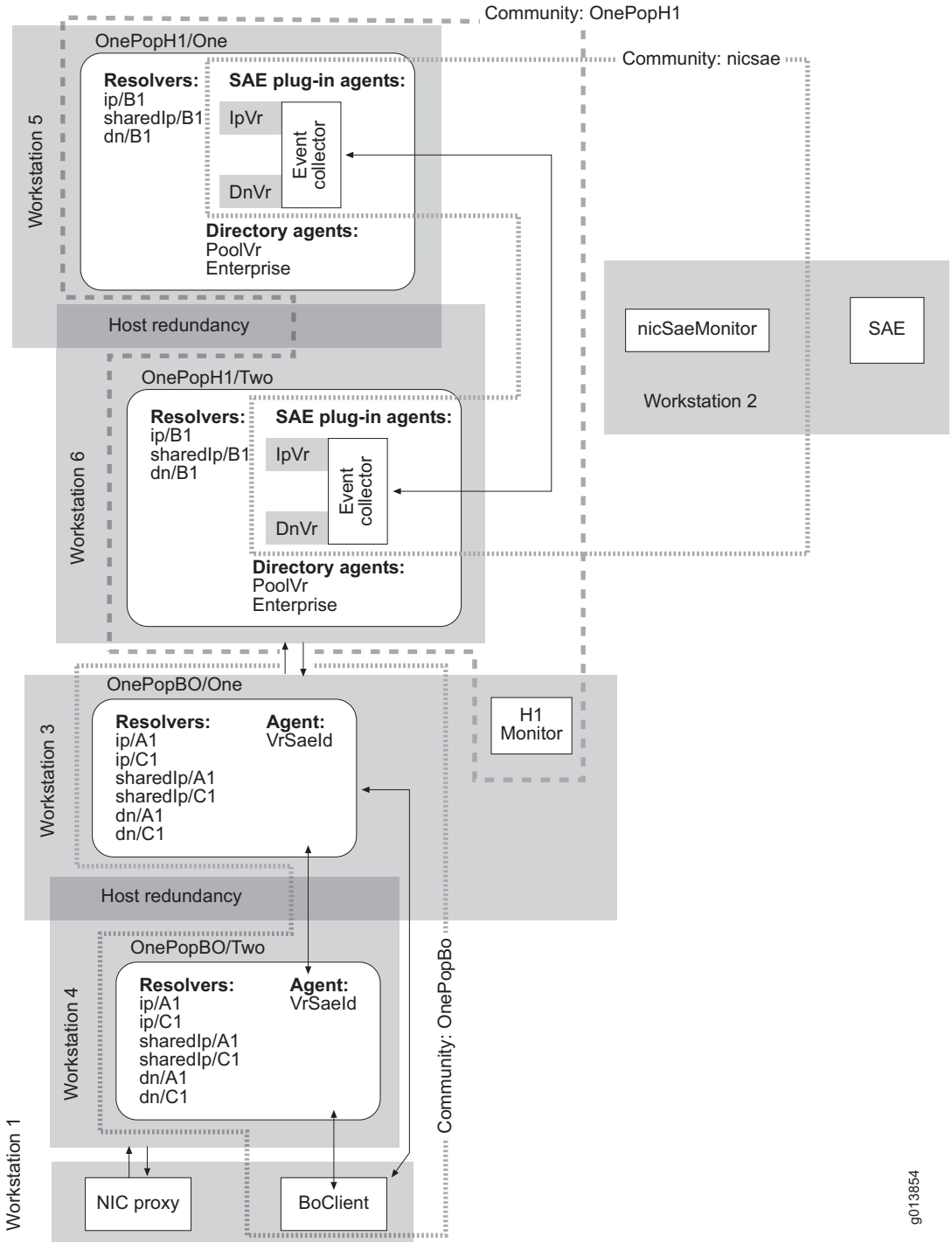
The redundant agents form a community called nicsae with monitor nicSaeMonitor, which tracks them. The redundant agents are identified in the community by the names OnePopH1/One and OnePopH1/Two; these names specify their hosts and provide unique identifiers for the redundant agents.

The redundant hosts OnePopBO/One and OnePopBO/Two are members of a community called OnePopBO. This community supports the monitor, BoClient, which is installed on the machine that supports the NIC proxy. BoClient tracks the connections between the redundant hosts OnePopBO/One and OnePopBO/Two from the point of view of the NIC client (NIC proxy).

Similarly, the redundant hosts OnePopH1/One and OnePopH1/Two are members of a community called OnePopH1. This community has one monitor, H1Monitor, which is located on the same machine as the SAE and tracks the connections among the redundant hosts in the same community, their primary host, and the other hosts in the configuration.

H1Monitor and nicSaeMonitor are part of the monitor process OnePop, which is also installed on the same machine as the SAE. BoClient is part of the monitor process OnePopClient, which is installed on the same machine as the NIC proxy.

Figure 42: OnePopAllRealms Distributed Configuration



MultiPop Scenario

The MultiPop scenario illustrates a configuration that involves two POPs: Montreal and Ottawa. This configuration does not provide redundancy. The NIC proxy communicates with the back office host (BackOffice), which in turn communicates with the POP hosts (MontrealHost and OttawaHost). Hosts MontrealHost and OttawaHost support equivalent hosts and agents and manage resolutions in the same way.

When host BackOffice receives a data key from the NIC proxy, the following sequence of events occurs:

1. Host BackOffice forwards requests as follows:
   - If the request is for the Montreal POP, host BackOffice forwards the request to POP host MontrealHost.
   - If the request is for the Ottawa POP, host BackOffice forwards the request to POP host OttawaHost.
2. Delegating tasks to other resolvers as necessary, the resolvers in the POP obtain data values that correspond to the data key request, and return them.
3. The POP host returns the data values to host BackOffice, which returns the value to the NIC proxy.

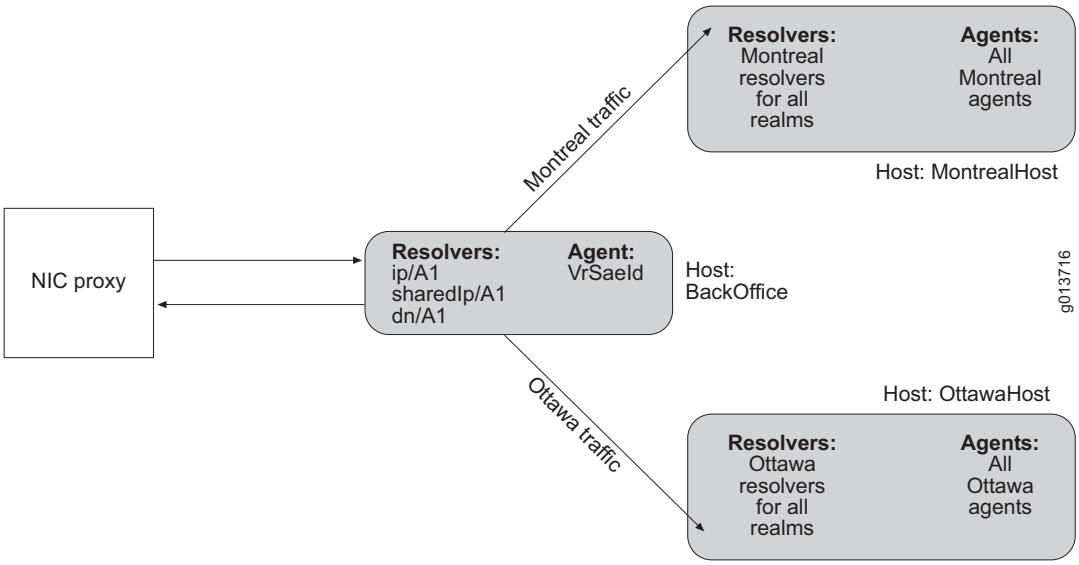
The scenario shows three realms for this configuration:

- IP
- Shared IP
- DN

Each realm provides a different type of resolution. The following sections provide information about these realms.

Figure 43 illustrates this configuration.

Figure 43: MultiPop Configuration



IP Realm

This realm accommodates the situation in which IP pools are configured locally on each VR. The resolution process takes a subscriber's IP address as the key and returns a reference to the SAE managing this subscriber as the value. This realm uses essentially the same resolution process as the ip realm for the OnePop scenario (see Figure 23 on page 320). However, some of the constraints differ.

The following agents interact with the resolvers in this realm:

- Directory agents montrealPoolVr and ottawaPoolVr collect and publish information that maps IP pools to VRs. Each agent publishes only the information that is relevant to its POP. You achieve selective publishing by relating an Ottawa scope to the VRs in the Ottawa POP and a Montreal scope to the VRs in the Montreal POP and defining a search filter for the agents to load only the VRs in its POP.
- Directory agent VrSaeId in the back office collects and publishes information that maps VRs to SAEs for both POPs.

When the NIC proxy sends a subscriber's IP address to host BackOffice, the following sequence of events occurs:

1. Host BackOffice passes the IP address to resolver ip/A1.
2. Resolver ip/A1 obtains an IP pool for the IP address.
3. Resolver ip/A1, based on the value of the IpPool, forwards the request to ip/B1montreal or ip/B1ottawa.

4. Resolver ip/B1montreal or resolver ip/B1ottawa obtains a VR name for this IP pool and returns the VR name to resolver ip/A1.

5. Resolver ip/A1 forwards the VR name to resolver ip/C1.

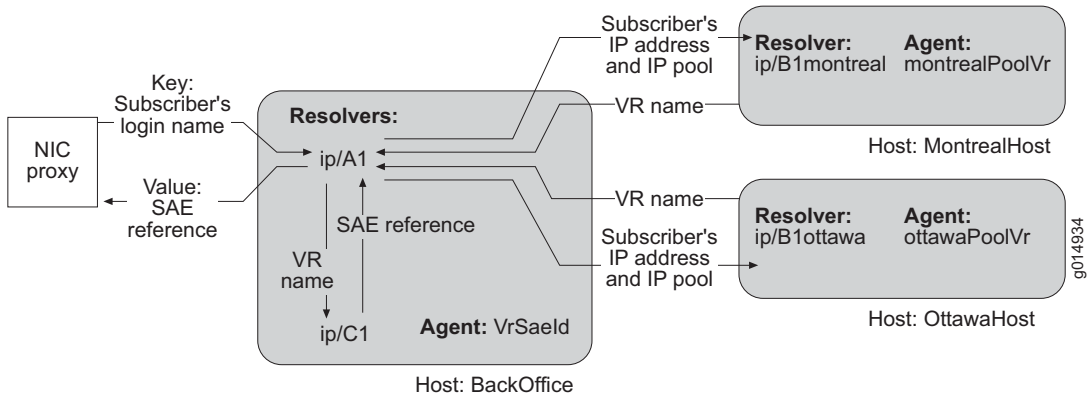
6. Resolver ip/C1 obtains the SAE identity for this VR and returns the value to resolver ip/A1.

7. Resolver ip/A1 returns the SAE reference to its host.

8. Host BackOffice returns the SAE reference to the NIC proxy.

Figure 44 illustrates the interactions of the NIC components for this realm.

Figure 44: IP Realm for MultiPop Configuration



Shared IP Realm

This realm accommodates the situation in which IP pools are shared by VRs in the same POP. The realm takes a subscriber's IP address as the key and returns the corresponding SAE as the value. Figure 24 on page 321 shows the resolution graph for this realm.

The following agents interact with resolvers in this realm:

- Directory agents montrealPoolVr and ottawaPoolVr collect and publish information about the mappings of IP pools to VRs. Each agent publishes only the information that is relevant to its POP.

- SAE plug-in agents montrealIpVr and ottawaIpVr collect and publish information about the mappings of subscriber IP addresses to VRs. Each agent publishes only the information that is relevant to its POP.

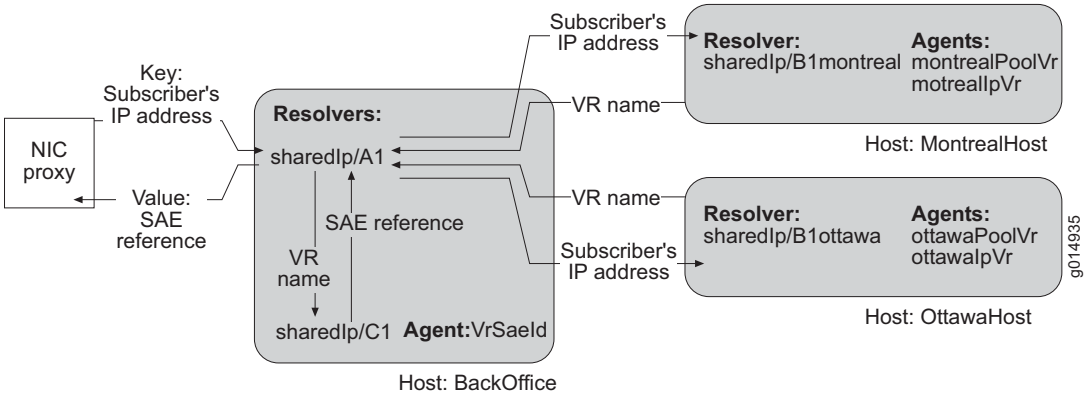
- Directory agent VrSaeId in the back office collects and publishes information about the mappings of VRs to SAEs for both POPs.

When the NIC proxy sends a subscriber's IP address to host BackOffice, the following sequence of events occurs:

1. Host BackOffice passes the IP address to resolver sharedIp/A1.
2. Resolver sharedIp/A1 obtains an IP pool for the IP address.
3. Resolver sharedIp/A1, based on the value of the IP pool, forwards the request to sharedIp/B1montreal or sharedIp/B1ottawa.
4. Resolver sharedIp/B1montreal or resolver sharedIp/B1ottawa obtains a VR name for this IP address and returns the VR name to resolver sharedIp/A1.
5. Resolver sharedIp/A1 forwards the VR name to resolver sharedIp/C1.
6. Resolver sharedIp/C1 obtains the SAE identity for this VR and returns the value to resolver sharedIp/A1.
7. Resolver sharedIp/A1 passes the SAE reference to its host.
8. Host BackOffice returns the SAE reference to the NIC proxy.

Figure 45 illustrates the interactions of the NIC components for this realm.

Figure 45: Shared IP Realm for MultiPop Configuration

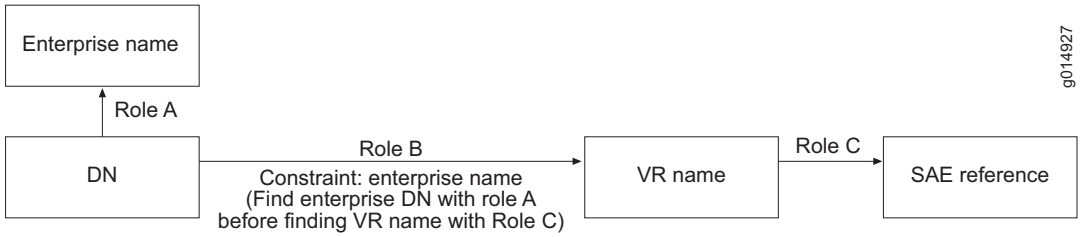


DN Realm

The DN realm takes the DN of an access subscriber (an access DN) as the key and returns the corresponding SAE as the value. Figure 46 shows the resolution process for this realm.

Figure 46 shows the resolution graph for this realm.

Figure 46: Resolution Graph for MultiPOP DN Realm



The following agents interact with resolvers in this realm:

- Directory agents ottawaEnterprise and montrealEnterprise collect and publish information about the DNs of enterprise subscribers (enterprise DNs). Each agent publishes only the information that is relevant to its POP. You achieve selective publishing by relating an Ottawa service scope to the enterprises in the Ottawa POP and a Montreal service scope to the enterprises in the Montreal POP and defining a search filter for the agents to load only the enterprises in its POP.

- SAE plug-in agents montrealDnVr and ottawaDnVr collect and publish information about the mappings of access DNs to VRs. Each agent publishes only the information that is relevant to its POP.

- Directory agent VrSaeId collects and publishes information about the mappings of VRs to SAEs for both POPs.

When the NIC proxy sends an access DN to host BackOffice, the following sequence of events occurs:

1. Host BackOffice passes the access DN to resolver dn/A1.

2. Resolver dn/A1 obtains an enterprise DN for the access DN.

3. Resolver dn/A1, based on the value of the enterprise DN, forwards the request to dn/B1montreal or dn/B1ottawa.

4. Resolver dn/B1montreal or resolver dn/B1ottawa obtains a VR name for this enterprise DN and returns the VR name to resolver dn/A1.

5. Resolver dn/A1 forwards the VR name to resolver dn/C1.

6. Resolver dn/C1 obtains the SAE reference for this VR and returns the value to resolver dn/A1.

7. Resolver dn/A1 passes the SAE reference to its host.

8. Host BackOffice returns the SAE reference to the NIC proxy.

Figure 47 illustrates the interactions of the NIC components for this realm.

Figure 47: DN Realm for MultiPop Configuration