Chapter 19

NIC Configuration Scenarios

This chapter provides detailed descriptions of the network information collector (NIC) configuration scenarios. Topics include:

- Overview of NIC Configuration Scenarios on page 238
- OnePop Scenario on page 238
- OnePopPcmm Scenario on page 242
- OnePopDynamicIp Scenario on page 244
- OnePopSharedIp Scenario on page 246
- OnePopStaticRouteIp on page 248
- OnePopAcctId Scenario on page 250
- OnePopLogin Scenario on page 252
- OnePopPrimaryUser on page 255
- OnePopDnSharedIp Scenario on page 257
- OnePopAllRealms Scenario on page 261
- MultiPop Scenario on page 265

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.

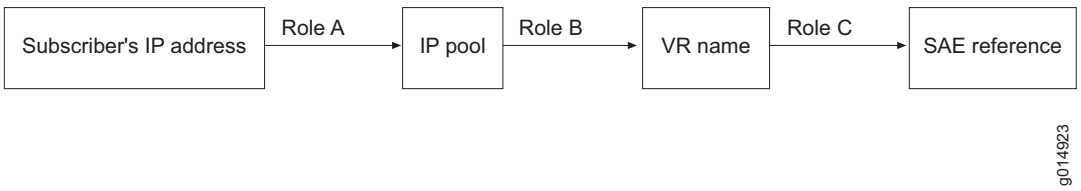
For a summary of the NIC configuration scenarios included in the sample data, see *Chapter 9, Locating Subscriber Information with the NIC*.

OnePop Scenario

The OnePop scenario illustrates a configuration that supports one POP. The realm for this configuration accommodates the situation in which IP address 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 12 shows the resolution graph for this realm.

Figure 12: 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 address 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 and a distributed configuration. The OnePop Centralized configuration also provides an example of NIC host redundancy.

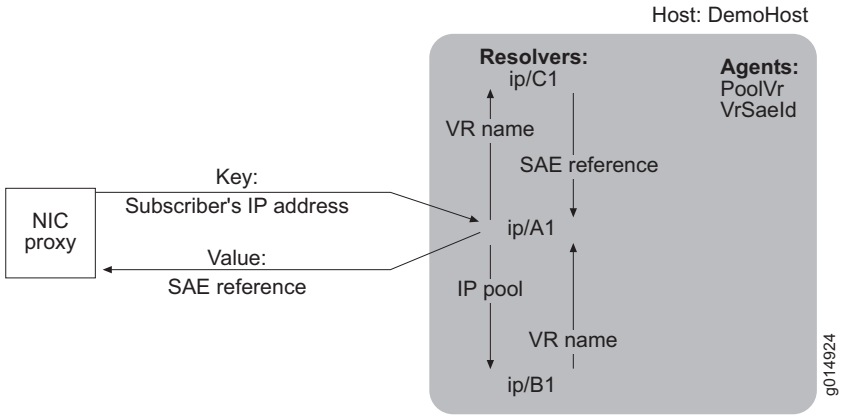
Centralized Configuration

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 13 shows the interactions of the NIC components for this realm.

Figure 13: OnePop Centralized Configuration

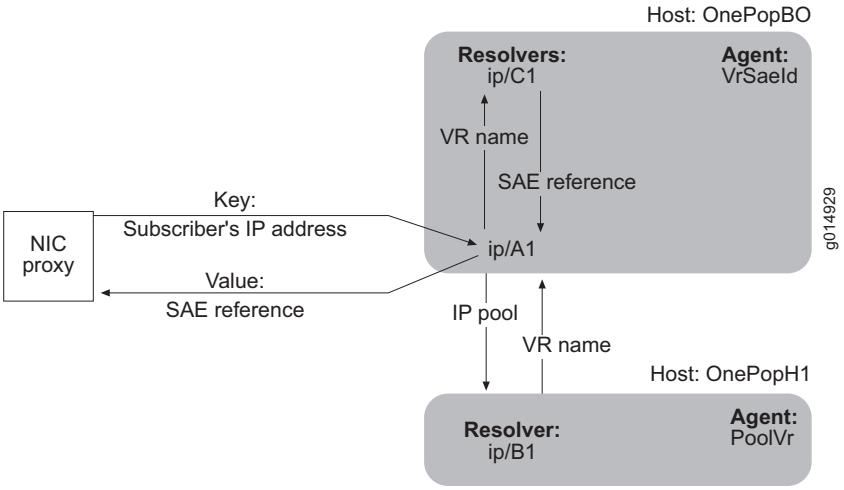


Distributed Configuration

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 239).

Figure 14 illustrates the interactions of the NIC components for this realm.

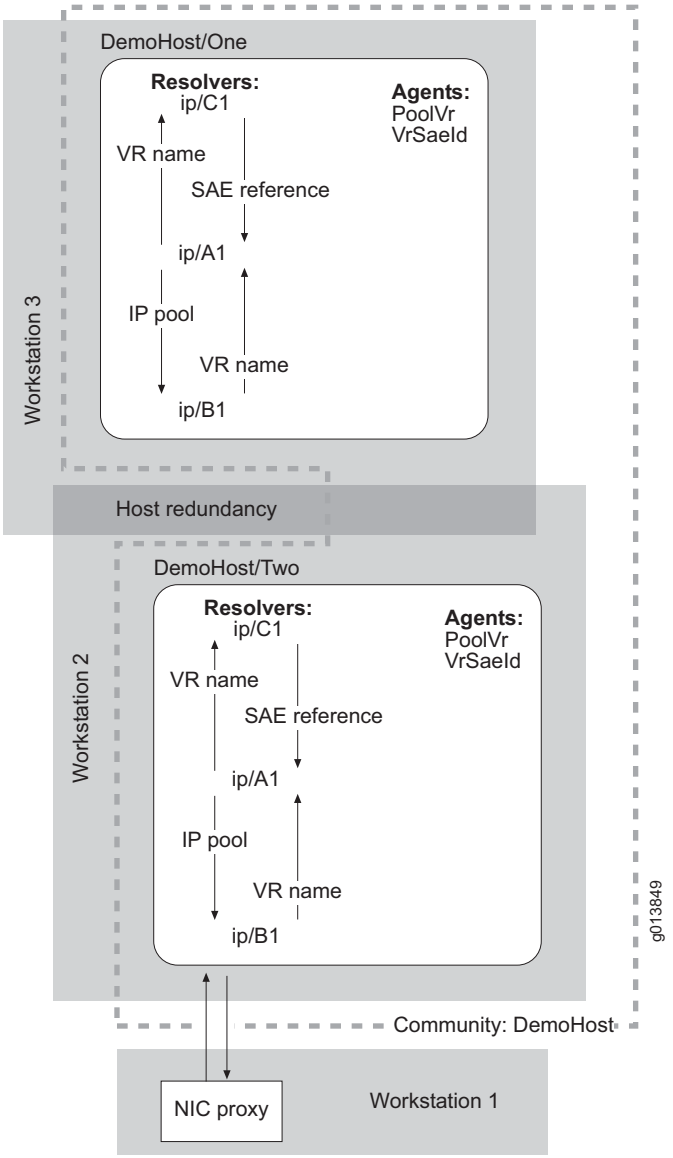
Figure 14: OnePop Distributed Configuration



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 15: Redundancy for OnePop Centralized Configuration



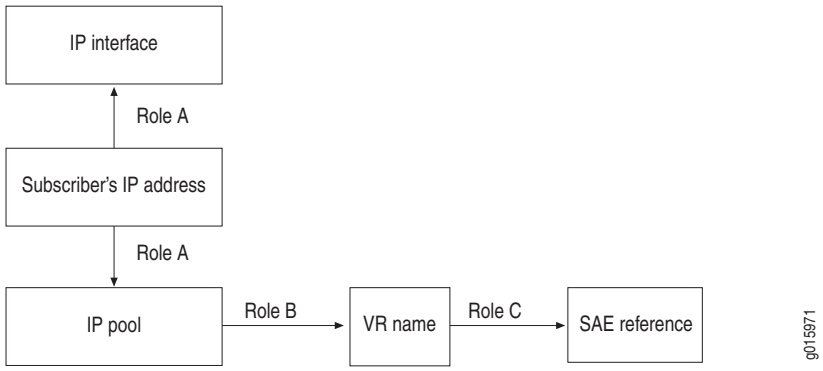
OnePopPcmm Scenario

This scenario is similar to the OnePop configuration scenario. It illustrates a configuration in which an assigned subscriber IP address managed by a network device such as a cable modem termination system (CMTS) device resolves to a reference to the SAE managing this subscriber. In this situation, the SAE acts as an application manager and interacts with the CMTS through a policy server.

The OnePopPcmm configuration scenario supports a PacketCable Multimedia Specification (PCMM) environment in which you use the assigned IP subscriber method to log in subscribers and in which you use the NIC to determine the subscriber's SAE. The realm for this configuration accommodates the situation in which IP pools are configured locally on each application manager group object. These IP pools represent an IP pools-managed policy decision point (PDP) group for one or more CMTS devices.

Figure 16 shows the resolution graph for this realm.

Figure 16: Resolution Process for Pcmm_am Realm



This scenario uses the same agents as the OnePop scenario. For the OnePopPcmm configuration scenario, the agent collects information from the application manager object instead of the virtual router entry. A virtual router name is generated in the format "default"@<pdpGroup>.

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

Centralized Configuration

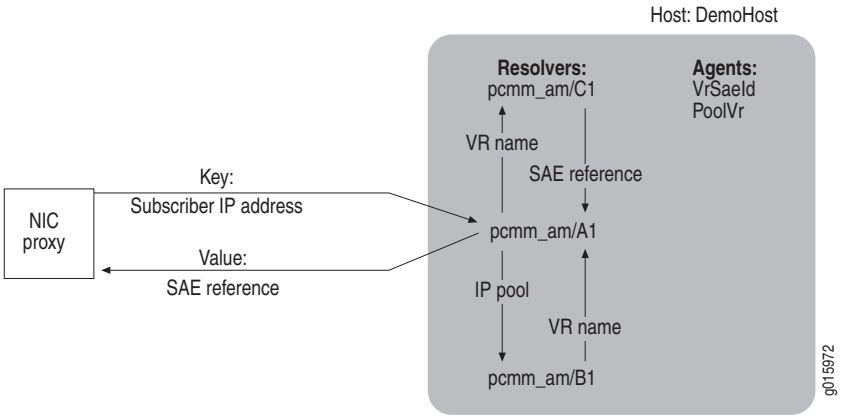
In this configuration, the single host DemoHost supports all agents and resolvers. When a NIC proxy sends a subscriber's IP address to host DemoHost, the following sequence of actions occurs:

1. The host passes an assigned subscriber IP address resolver A1.
2. Resolver A1 obtains the IP pool name and the interface name, and forwards the request to resolver B1.
3. Resolver B1 obtains the 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 it 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 17 show the interactions of the NIC components for this realm.

Figure 17: OnePopPcmm Centralized Configuration

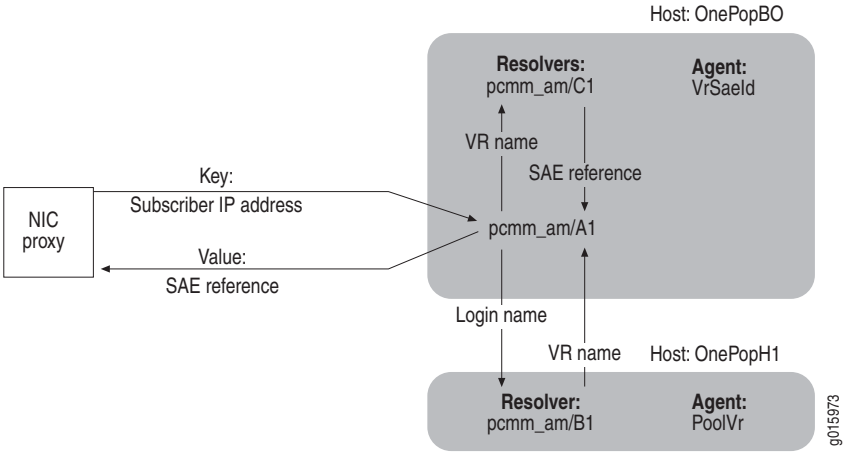


Distributed Configuration

In this configuration, the agents and resolvers are distributed among two 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 242).

Figure 18 illustrates the interactions of the NIC components for this realm.

Figure 18: OnePopPcmm Distributed Configuration



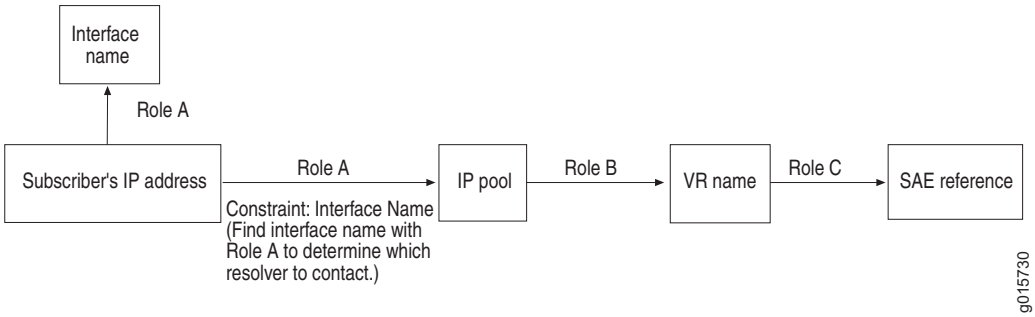
OnePopDynamicIp Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario. The realm for this configuration accommodates the situation in which IP address 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. In this scenario, the SAE acts as a combined application manager and policy server; it directly manages CMTS devices.

Figure 19 shows the resolution graph for this realm.

Figure 19: Resolution Process for dynamicIp Realm



The following agents collect information for resolvers in this realm:

- Directory agent PoolVr collects and publishes information about the mappings of IP address 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.

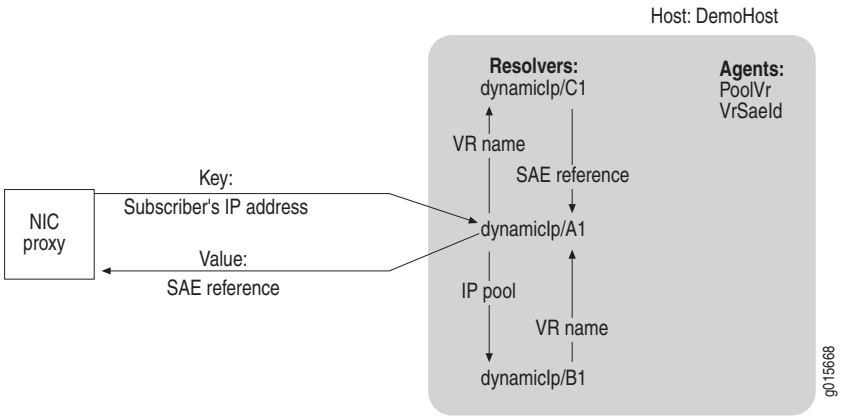
5. Resolver C1 obtains an SAE reference for the VR and returns the VR identity to resolver A1.

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

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

Figure 20 illustrates the interactions of the NIC components for this realm.

Figure 20: OnePopDynamicIp Centralized Configuration

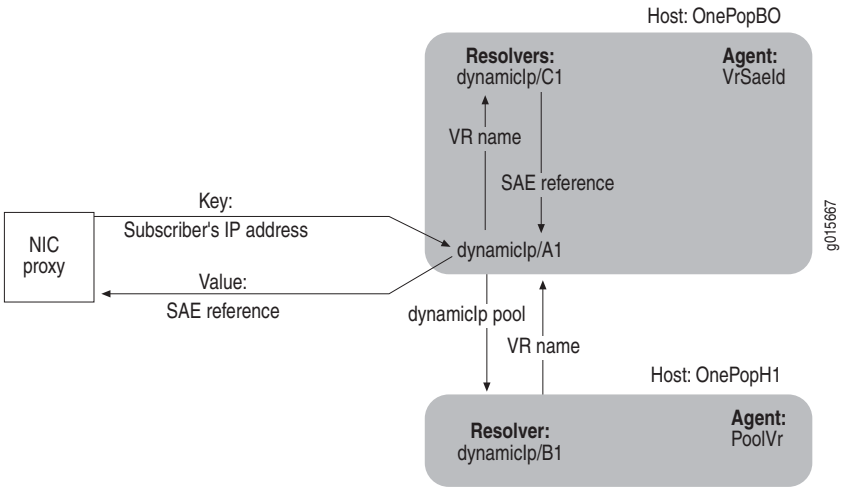


Distributed Configuration

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 244).

Figure 21 illustrates the interactions of the NIC components for this realm.

Figure 21: OnePopDynamicIp Distributed Configuration

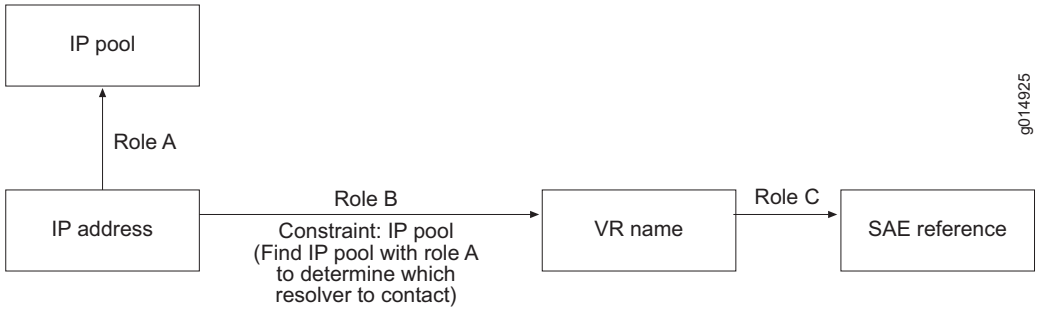


OnePopSharedIp Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario. However, the realm for this configuration accommodates the situation in which IP address 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 22 shows the resolution graph for this realm.

Figure 22: 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 address pools used by the VRs in a POP. Because the IP address 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.

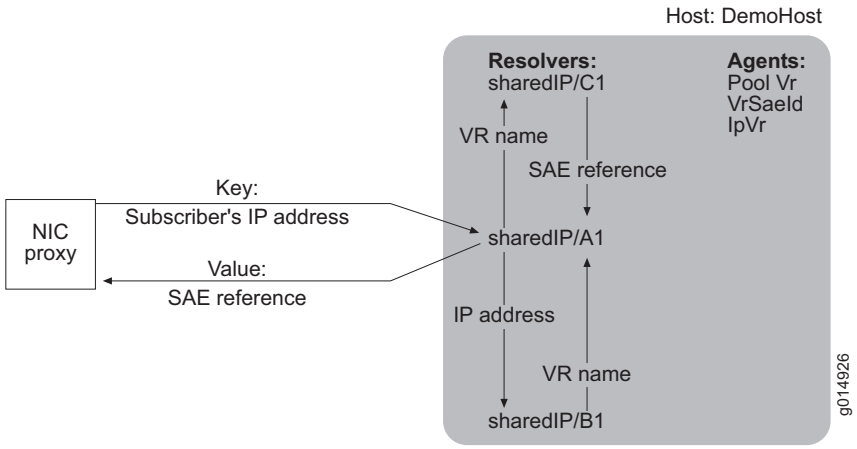
6. Resolver C1 obtains an SAE reference for the VR and returns the SAE reference to resolver A1.

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

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

Figure 23 shows the interactions of the NIC components for this realm.

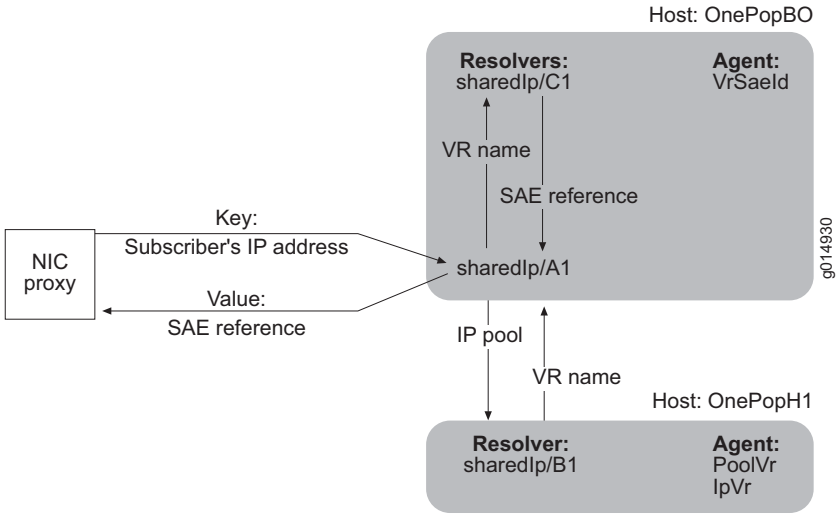
Figure 23: OnePopSharedIP Centralized Configuration



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 24 illustrates the interactions of the NIC components for this realm.

Figure 24: OnePopSharedIP Distributed Configuration

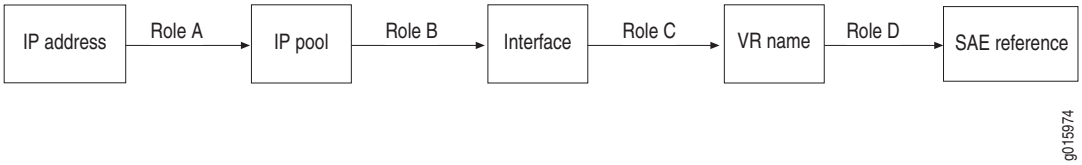


OnePopStaticRouteIp

The OnePopStaticRouteIp configuration scenario for NIC resolves an assigned IP address for a subscriber whose traffic enters the network through an interface on a JUNOS routing platform to a reference for the SAE that manages the interface. The realm for this configuration accommodates the situation in which the network publisher component gathers interface information for the JUNOS routing platforms. The resolution process takes a subscriber's IP address as a key and returns a reference to the SAE that manages the interface.

Figure 25 shows the resolution graph for this realm.

Figure 25: Resolution Process for the StaticRouteIp Realm



The following agents collect information for resolvers in this realm:

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

The agents obtain information from the interfaceConfiguration attribute of the EdgeRouter entry in the directory and read an XML document that conforms to the networkConfig.xsd schema. If this scenario is used with a different router type, you can edit the XML document.

For information about the XML document, see *Chapter 12, Obtaining Interface Configuration for OnePopStaticRouteIp on Solaris Platforms*.

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

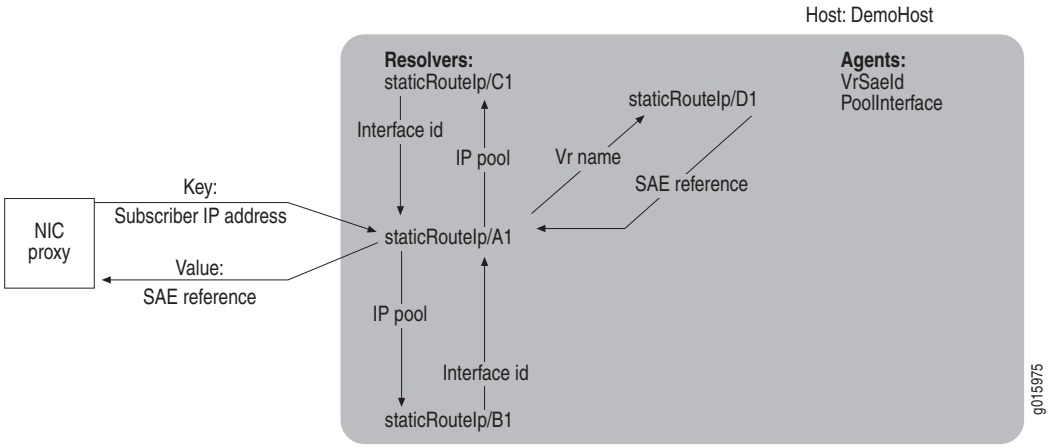
Centralized Configuration

In this configuration, the 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 obtains an IP pool for the IP address.
3. Resolver AI forwards the IP pool name to Resolver B1.
4. Resolver B1 obtains the interface ID for the IP pool and returns this value to resolver A1.
5. Resolver A1 forwards the interface ID to Resolver C1.
6. Resolver C1 resolves the interface ID to the VR name and returns the VR name to resolver A1.
7. Resolver A1 forwards the VR name to resolver D1.
8. Resolver D1 obtains a reference for the SAE managing the VR and returns the SAE reference to resolver A1.
9. Resolver A1 passes the SAE reference to its host.
10. The host returns the SAE reference to the NIC proxy.

Figure 26 shows the interactions of the NIC components for this realm.

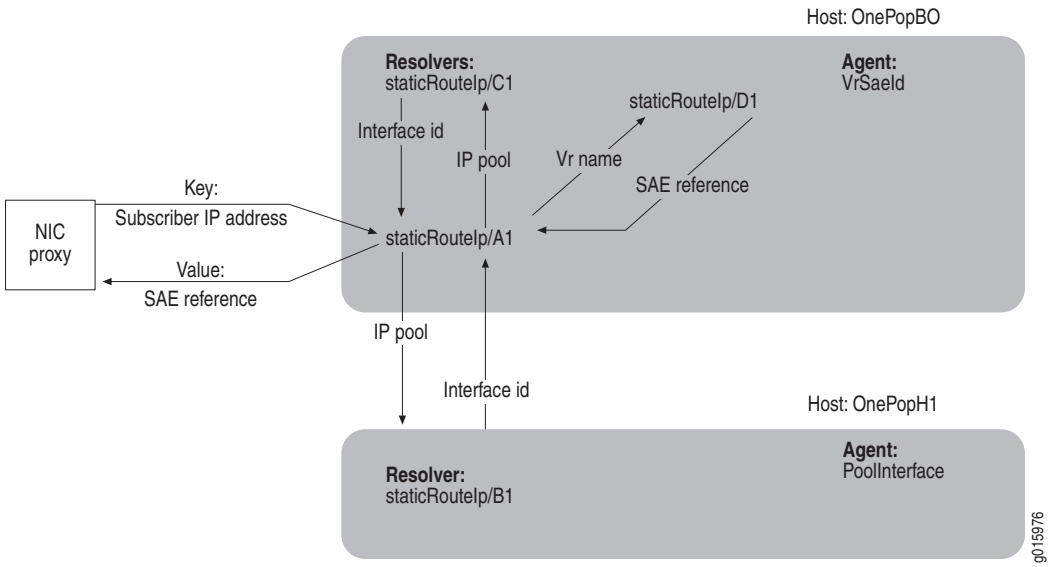
Figure 26: OnePopStaticRouteIp Centralized Configuration



Distributed Configuration

In this configuration, the agents and resolvers are distributed among two hosts. When a NIC proxy sends a subscriber IP address to host OnePopBO, the resolvers execute the same actions as they do in the centralized configuration. Figure 27 illustrates the interactions of the NIC components for this realm.

Figure 27: OnePopStaticRouteIp Distributed Configuration

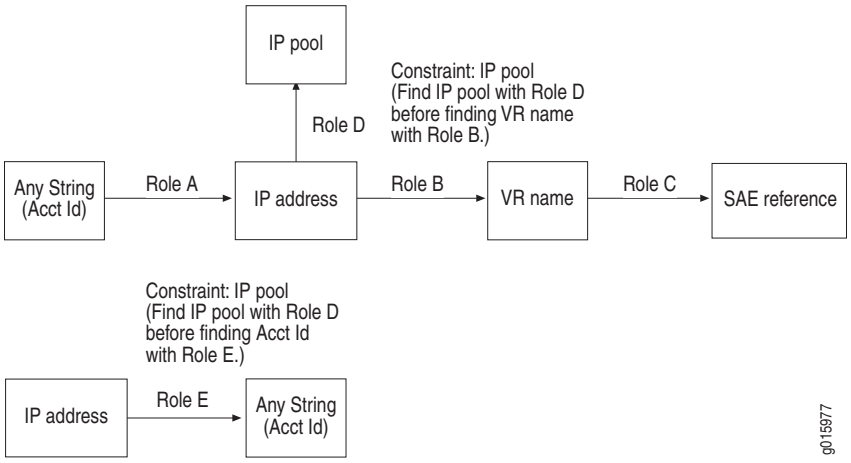


OnePopAcctId Scenario

This scenario illustrates a configuration in which subscribers have an accounting ID, as defined by the LDAP attribute accountingUserId or the plug-in attribute PA_ACCOUNTING_ID. The realms for this configuration accommodate two independent resolution processes, which can be used by the SRC Volume-Tracking Application (SRC-VTA).

Figure 28 shows the resolution graphs for this realm.

Figure 28: Resolution Process for acctId Realm



The following agents collect information for resolvers in this realm:

- Directory agent PoolVr collects and publishes information about the mappings of IP address pools to VRs.
- Directory agent VrSaeId collects and publishes information about the mappings of virtual routers and the mappings between virtual routers and SAEs.
- SAE plug-in agent AcctIdIp collects and publishes information about the mappings of accounting IDs of subscribers to subscriber IP addresses.
- SAE plug-in agent IpAcctId collects and publishes information about the mappings of subscriber IP addresses to accounting IDs.

The OnePopAcctId scenario provides one host for a centralized configuration. In this configuration the single host DemoHost supports all agents and resolvers. Two NIC proxies are associated with the configuration. One NIC proxy (called acct-sae in this description) submits accounting IDs, and another NIC proxy (called addr-acct in this description) submits subscribers' IP addresses.

When the NIC proxy sends an accounting ID to host DemoHost, the following sequence of events occurs:

1. The host passes the subscriber's accounting ID to resolver A1.
2. Resolver A1 obtains an IP address for the account ID.
3. Resolver A1 forwards the IP address to Resolver D1.
4. Resolver D1 obtains the IP pool for the IP address and returns it to Resolver A1.
5. Resolver A1 forwards the IP address and IP pool to Resolver B1.
6. Resolver B1 obtains the VR name and return it to resolve A1.

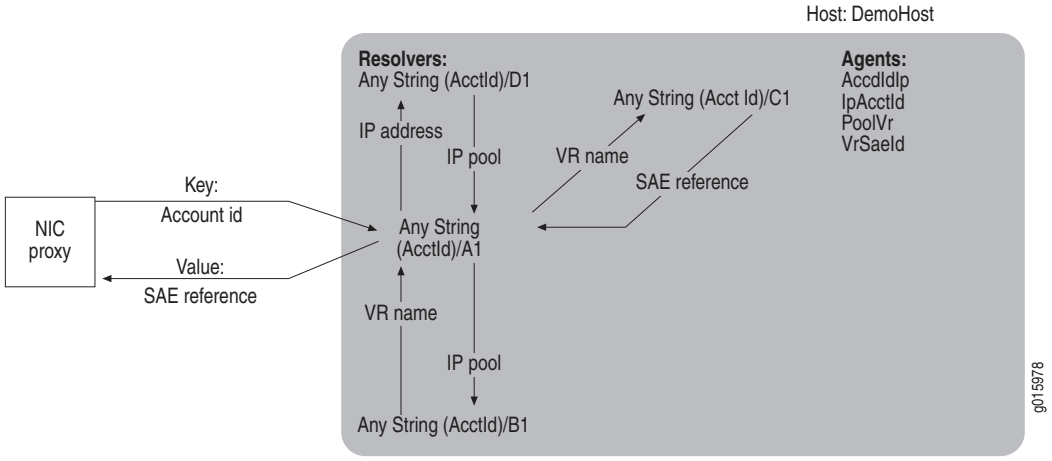
7. Resolver A1 forwards the VR name to resolver C1.
8. Resolver C1 obtains the SAE reference for the VR name and returns it to resolver A1.
9. Resolver A1 passes the SAE reference to its host.
10. The host returns the SAE reference to the NIC proxy acct-sae.

When the NIC proxy sends an 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 forwards the IP address to resolver D1.
3. Resolver D1 obtains the IP pool for the IP address and returns it to resolver A1.
4. Resolver A1 forwards the IP address and IP pool to resolver C1.
5. Resolver C1 obtains the accounting ID for the IP address and associated IP pool and returns the accounting Id to resolver A1.
6. Resolver A1 passes the accounting ID to its host.
7. The host returns the accounting ID to the NIC proxy addr-acct.

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

Figure 29: OnePopAcctId Centralized Configuration

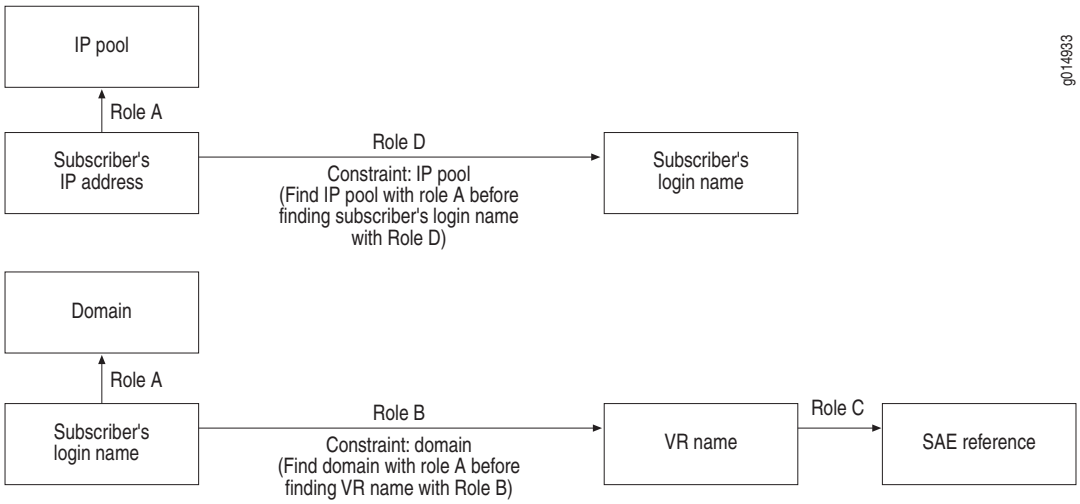


OnePopLogin Scenario

This scenario illustrates a configuration that is very similar to the OnePop scenario. The realm for this configuration accommodates two independent resolution processes, which are used by the SRC Volume Tracking Applications (SRC-VTAs) and may be used for other purposes.

Figure 30 shows the resolution graphs for this realm.

Figure 30: Resolution Processes login 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 address pools used by the VRs in a POP. The agent uses the information about the IP address 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.

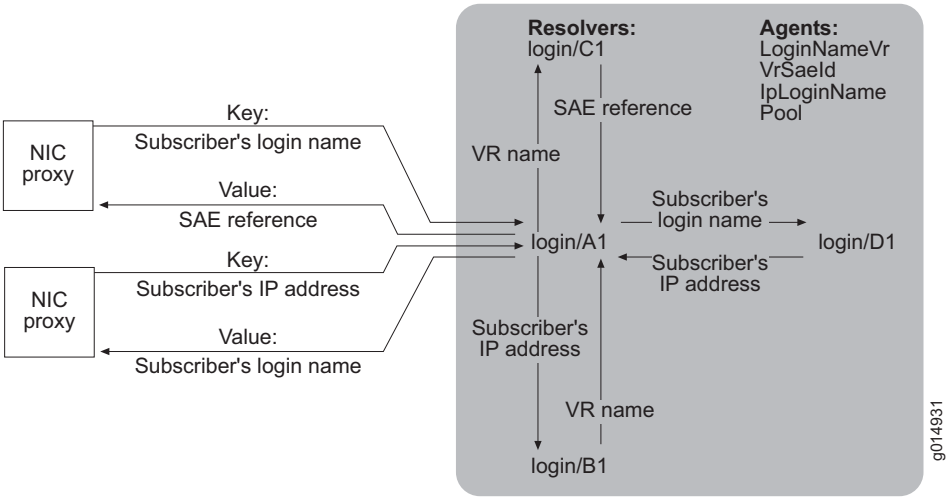
4. Resolver D1 obtains a login name for the IP address and returns the login name to resolver A1.

5. Resolver A1 passes the login name to its host.

6. The host returns the login name to the NIC proxy.

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

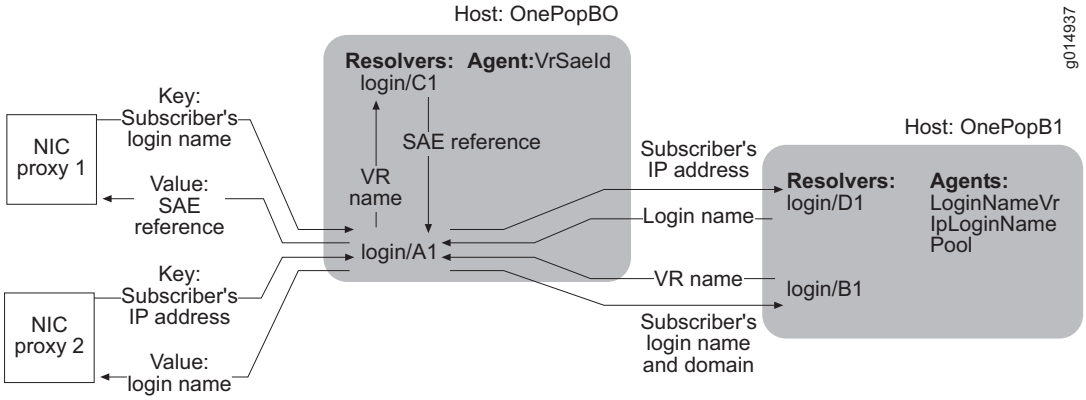
Figure 31: OnePopLogin Centralized Configuration



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 32 illustrates the interactions of the NIC components for this realm.

Figure 32: OnePopLogin Distributed Configuration

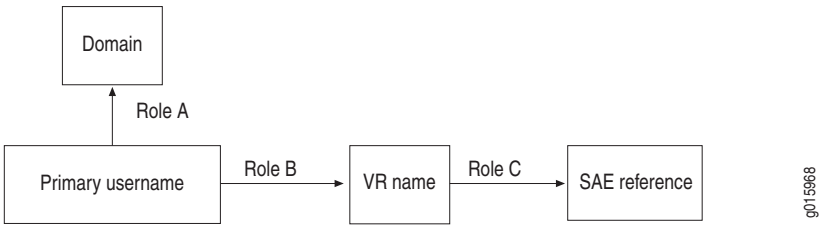


OnePopPrimaryUser

The OnePopPrimaryUser scenario is similar to one of the resolutions in the OnePopLogin scenario. In the OnePopPrimaryUser scenario, subscriber primary username, as identified by the PA_PRIMARY_USER_NAME attribute, is resolved to a reference for a managing SAE. The realm for this configuration accommodates a situation in which a NIC proxy provides a primary username.

Figure 33 show the resolution graph for this realm.

Figure 33: Resolution Processes for primary_user Realm



The following agents interact with resolvers in this realm:

- Directory agent VrSaeId collects and publishes information about virtual routers and the mappings between virtual routers and SAEs.
- SAE plug-in agent UserNameVr collects and publishes information about the mappings of subscriber primary usernames to VR names.

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

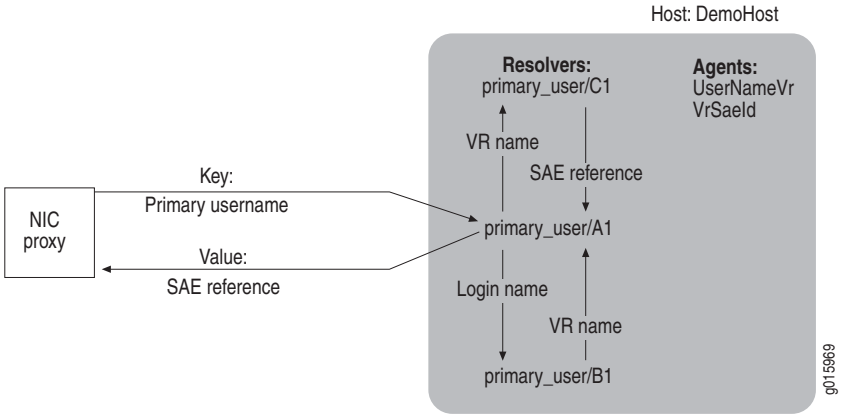
Centralized Configuration

In this configuration, a single host called DemoHost supports all agents and resolvers. When a NIC proxy send a subscriber's primary username to host Demo Host, the following sequence of events occurs:

1. The host passes the primary username to resolver A1.
2. (Optional) Resolver A1 resolves the primary username to its domain.
3. Resolver A1 forwards the primary username to resolver B1.
4. Resolver B1 obtains the name of the VR associated with the subscriber's primary username and returns the VR to resolver A1.
5. Resolver A1 forwards the VR to resolver C1.
6. Resolver C1 obtains the SAE reference for the SAE managing the VR and returns the SAE reference to resolver A1.
7. Resolver A1 returns the SAE reference to the host.
8. The host returns the SAE reference to the NIC proxy.

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

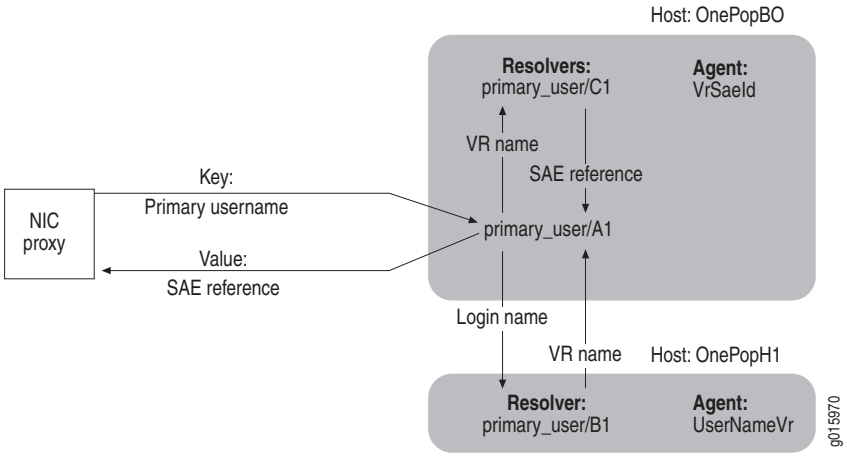
Figure 34: OnePopPrimaryUser Centralized Configuration



Distributed Configuration

In this configuration, the agents and resolvers are distributed among two hosts, When a NIC proxy sends a subscriber's primary username 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: OnePoIpPrimaryUser 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 22 on page 246).

- DN realm

  This realm uses essentially the same resolution process as the MultiPop DN realm (see Figure 43 on page 269). 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 36 on page 258 illustrates the centralized configuration, and Figure 37 on page 260 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 261.

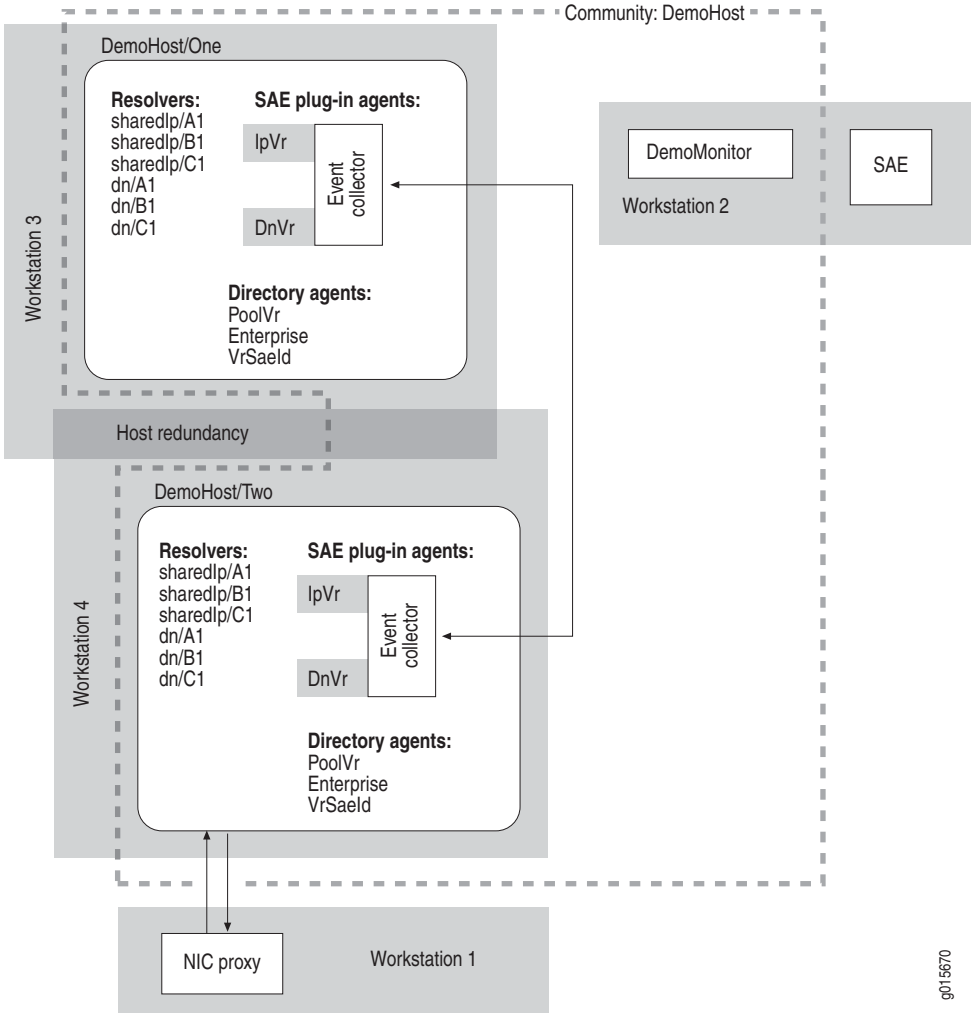
The OnePopAllRealms illustrates SAE plug-in agents configured to use SAE plug-in redundancy rather than SAE plug-in agents.

Centralized Configuration

Figure 36 on page 258 shows the centralized configuration for the scenario. Host DemoHost supports all resolvers and agents. The two SAE plug-in agents, IpVr and DnVr, share an event collector. Both plug-in agents have state synchronization enabled.

DemoHost is also 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.

Figure 36: OnePopDnSharedIp Realms Centralized Configuration



Distributed Configuration

Figure 37 on page 260 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. Host OnePopH1 supports one resolver for each realm and agents that are used by different realms.

Both hosts also have a redundant configuration. The redundant hosts for OnePopBO (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.

The redundant hosts for OnePopH1 (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.

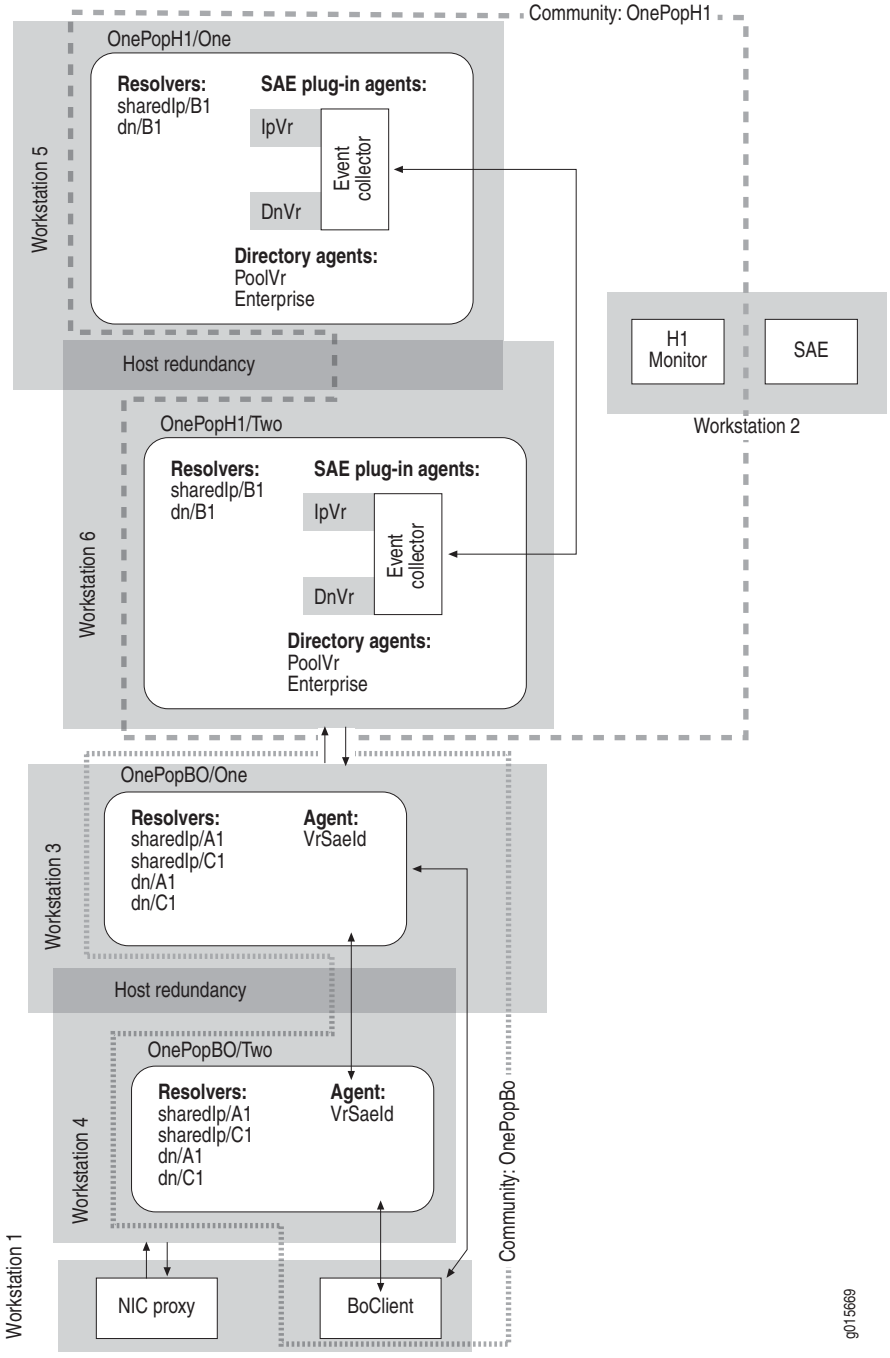
However, host OnePopH1 also supports two SAE plug-in agents, IpVr and DnVr, which share an event collector. These agents have state synchronization enabled.

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 37: 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 12 on page 238). However, some of the constraints differ.

- Shared IP

  The resolution process is identical to that for the OnePopShared scenario (see Figure 22 on page 246).

- DN realm

  This realm uses essentially the same resolution process as the MultiPop DN realm (see Figure 43 on page 269). 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 38 and Figure 39, you can determine exact pictures of the DN realms for the centralized and distributed configurations.

Centralized Configuration

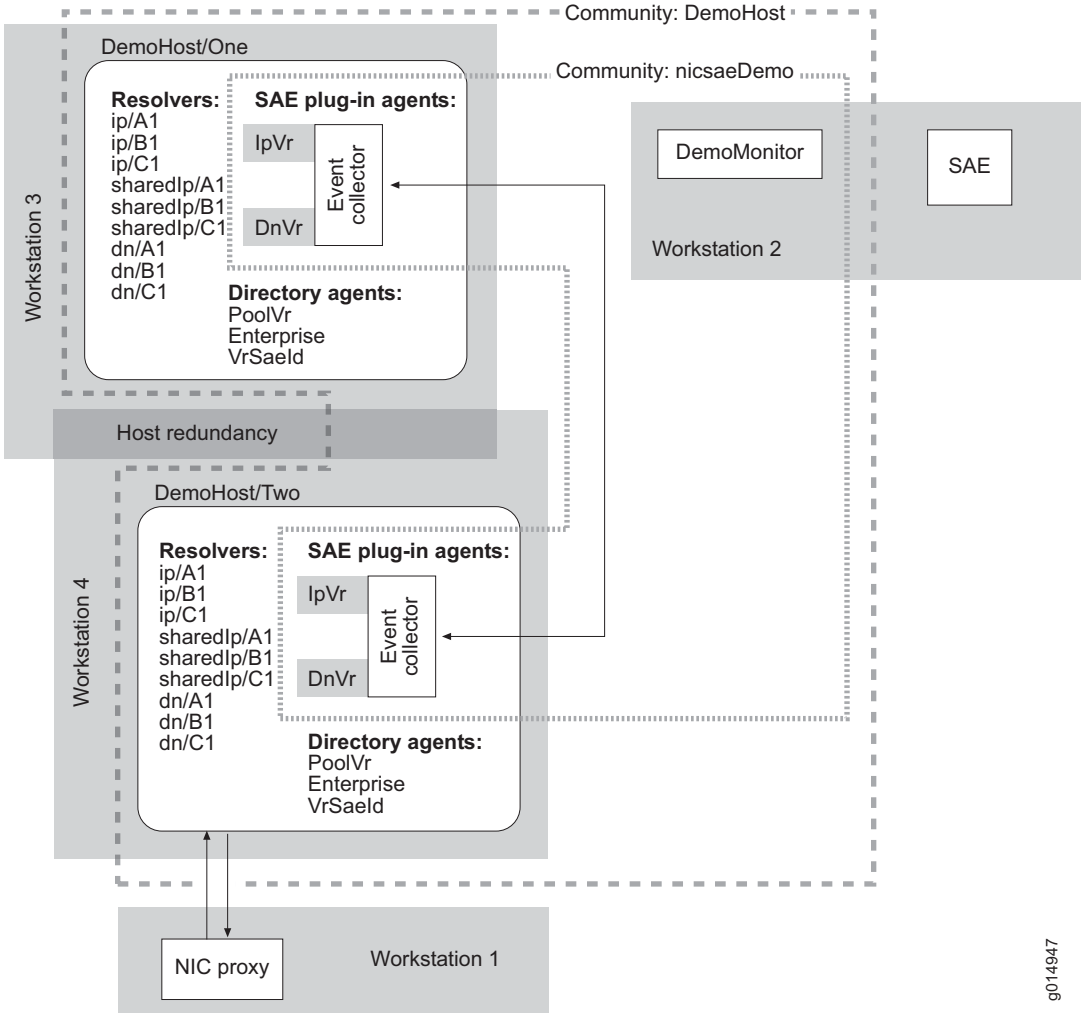
Figure 38 on page 262 shows the centralized configuration for the scenario. Host DemoHost supports all resolvers and agents. However, because host DemoHost is configured for redundancy, its redundant hosts (DemoHost/One and DemoHost/Two) perform the host function. The redundant hosts are on different 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 38: OnePopAllRealms Centralized Configuration



Distributed Configuration

Figure 39 on page 264 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.

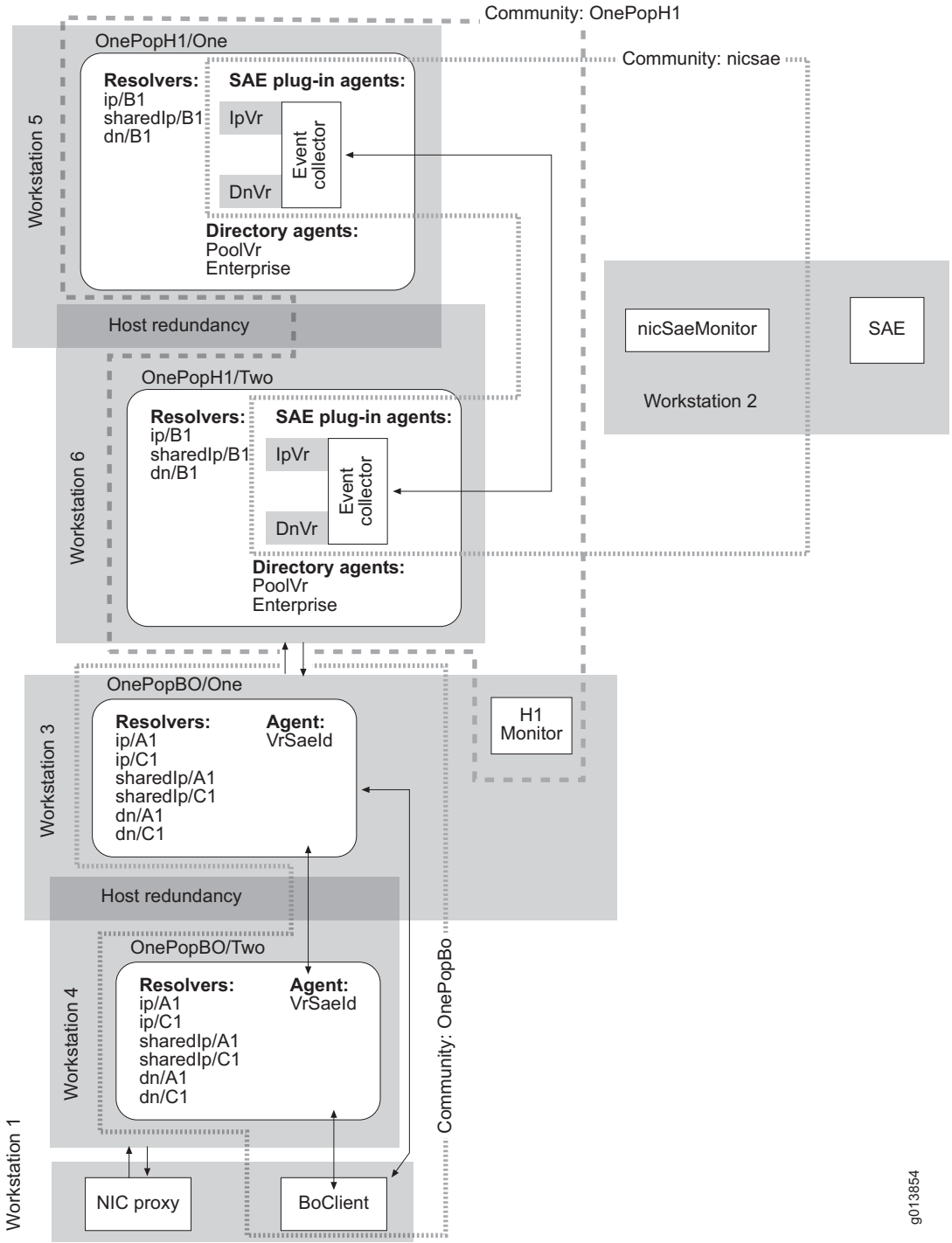
The redundant agents form a community called nicsae with monitor nicSaeMonitor, which tracks them. The redundant agents are identified in the community by the names OnePopH1/One and OnePopH1/Two; these names specify their hosts and provide unique identifiers for the redundant agents.

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 39: 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.

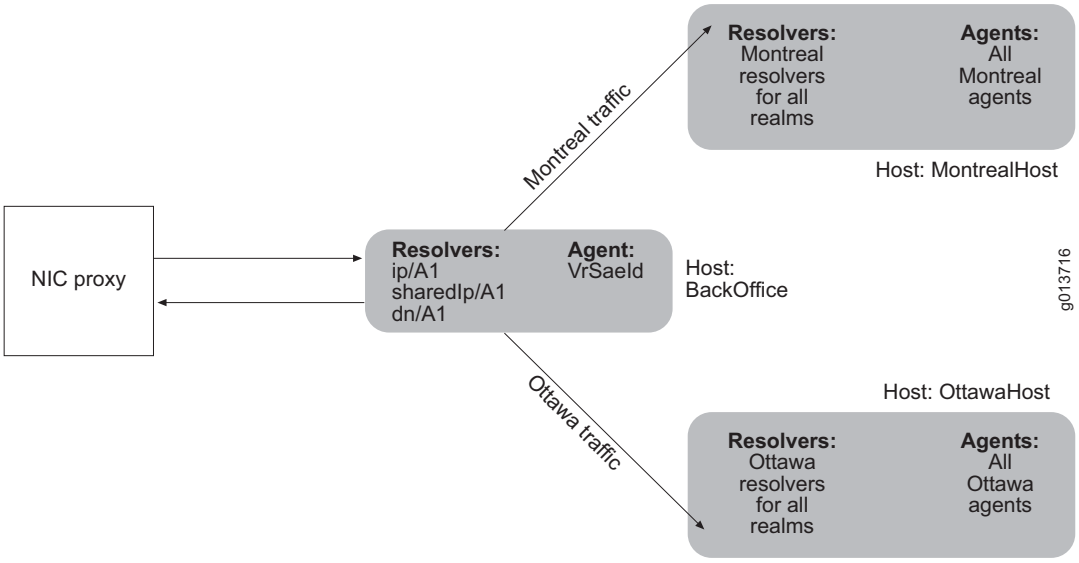
The scenario shows three realms for this configuration:

- IP
- Shared IP
- DN

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

Figure 40 illustrates this configuration.

Figure 40: MultiPop Configuration



IP Realm

This realm accommodates the situation in which IP address 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 12 on page 238). 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 address 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.

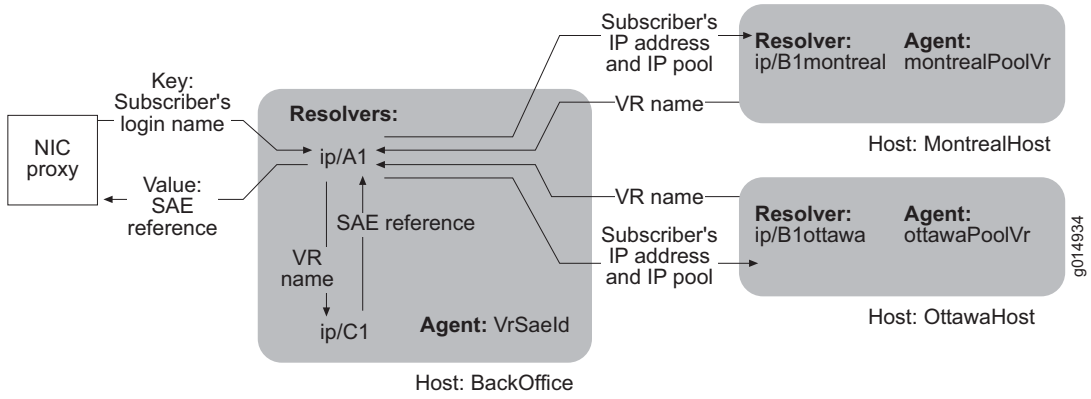
6. Resolver ip/C1 obtains the SAE identity for this VR and returns the value to resolver ip/A1.

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

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

Figure 41 illustrates the interactions of the NIC components for this realm.

Figure 41: iP Realm for MultiPop Configuration



Shared IP Realm

This realm accommodates the situation in which IP address 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 13 on page 239 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 address 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.

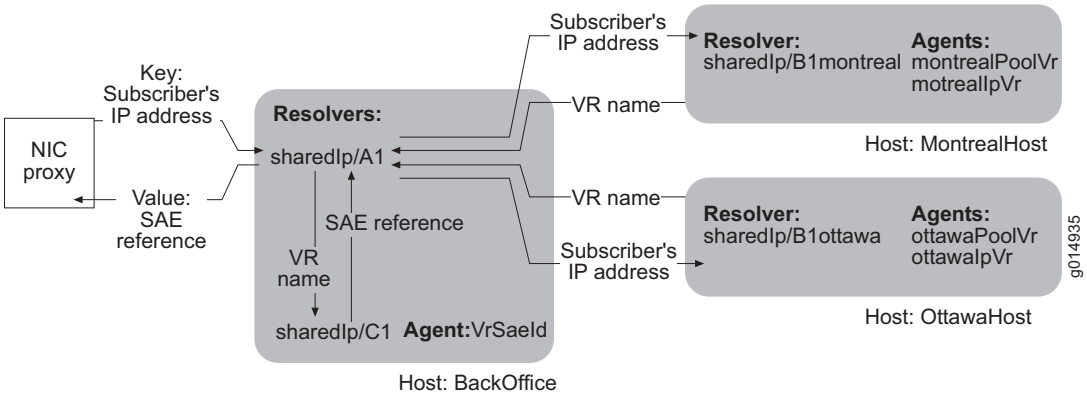
- Directory agent VrSaeId in the back office collects and publishes information about the mappings of VRs to SAEs for both POPs.

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 42 illustrates the interactions of the NIC components for this realm.

Figure 42: sharedIP Realm for MultiPop Configuration

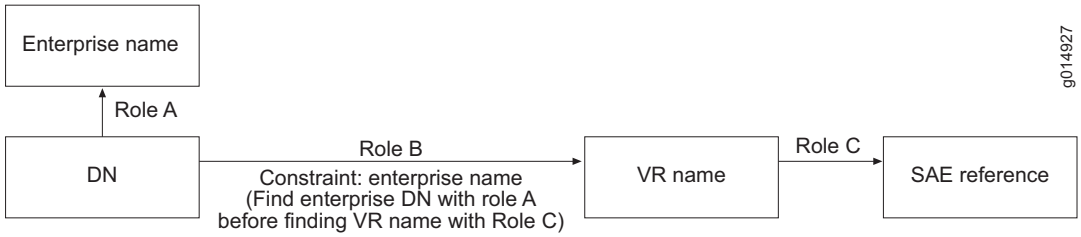


DN Realm

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 43 shows the resolution process for this realm.

Figure 43 shows the resolution graph for this realm.

Figure 43: 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 44 illustrates the interactions of the NIC components for this realm.

Figure 44: dn Realm for MultiPop Configuration