APM-DHCP Access Policy Example and Detailed Instructions (v3d)

Prepared by Mark Quevedo, f5 SSE, 2018–04–19a

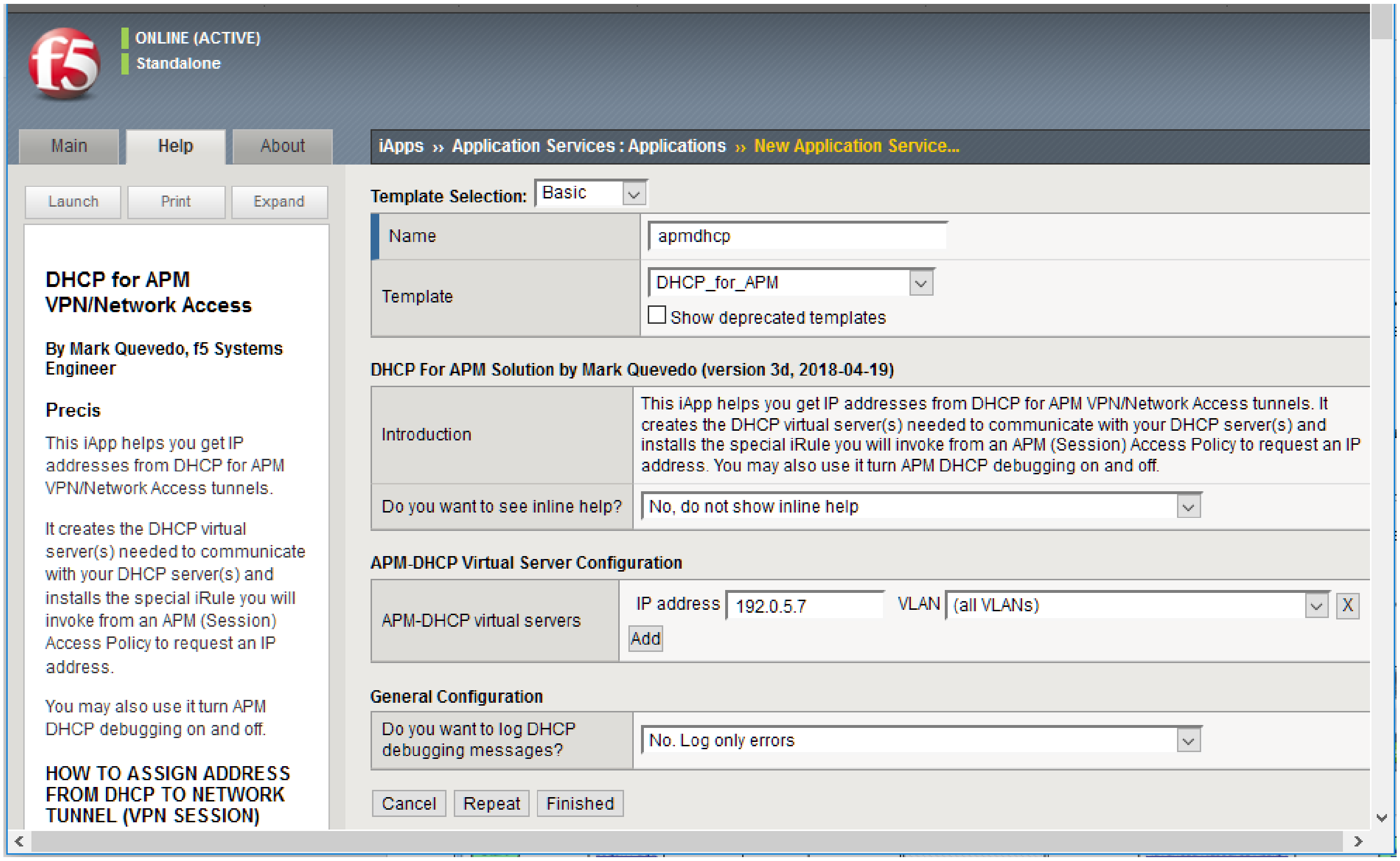
Ordinarily you assign an IP address to the "inside end" of an APM Network Tunnel (full VPN connection) from an address Lease Pool, from a static list, or from an LDAP or RADIUS attribute.

However, you might wish to assign an IP address obtained from a DHCP server. Perhaps the DHCP server owns all available client addresses. Perhaps it handles dynamic DNS for named client workstations. Or perhaps the DHCP server assigns certain users specific IP addresses, which may be referenced in ACL's elsewhere in the network.

f5's [old Firepass VPN supported DHCP address assignment]() but APM does not. We will use f5 iRules to enable DHCP with APM. We will send data from APM session variables to the DHCP server so it can issue the "right" IP address to each VPN tunnel based on user identity, client info, *etc*.

First install the APM-DHCP iApp template (file `DHCP_for_APM.tmpl`). Create a new Application Service as shown (choose any name you wish). Use the iApp to manage the APM-DHCP virtual servers you need. The iApp will also install the necessary iRules.

You must define at least one APM-DHCP virtual server to receive and send DHCP packets. Usually an APM-DHCP virtual server needs an IP address on the subnet on which you expect your DHCP server(s) to assign client addresses. You may define additional APM-DHCP virtual servers to request IP addresses from DHCP on additional subnets. However, if your DHCP server(s) support subnet-selection (see `session.dhcp.subnet` below) then you may need only a single APM-DHCP virtual server and it can use any IP that can talk to your DHCP server(s).



DHCP for APM VPN/Network Access

By Mark Quevedo, f5 Systems Engineer

Precis

This iApp helps you get IP addresses from DHCP for APM VPN/Network Access tunnels.

It creates the DHCP virtual server(s) needed to communicate with your DHCP server(s) and installs the special iRule you will invoke from an APM (Session) Access Policy to request an IP address.

You may also use it turn APM DHCP debugging on and off.

HOW TO ASSIGN ADDRESS FROM DHCP TO NETWORK TUNNEL (VPN SESSION)

Template Selection: Basic

Name	apmdhcp
Template	DHCP_for_APM ☐ Show deprecated templates

DHCP For APM Solution by Mark Quevedo (version 3d, 2018-04-19)

Introduction	This iApp helps you get IP addresses from DHCP for APM VPN/Network Access tunnels. It creates the DHCP virtual server(s) needed to communicate with your DHCP server(s) and installs the special iRule you will invoke from an APM (Session) Access Policy to request an IP address. You may also use it turn APM DHCP debugging on and off.
Do you want to see inline help?	No, do not show inline help

APM-DHCP Virtual Server Configuration

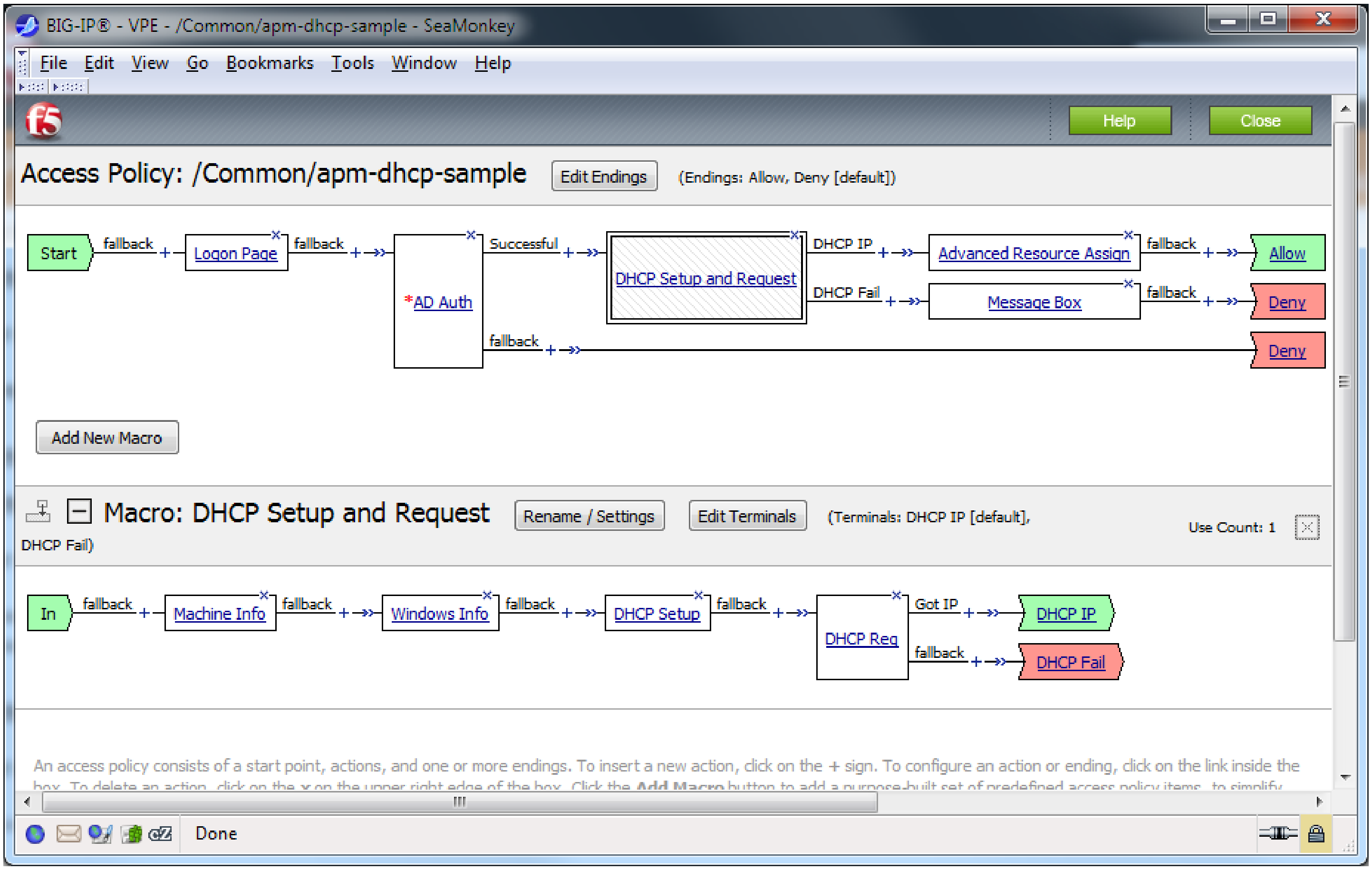
APM-DHCP virtual servers	IP address 192.0.5.7 VLAN (all VLANs)

General Configuration

Do you want to log DHCP debugging messages?	No. Log only errors

It is best to give each APM-DHCP virtual server a unique IP address but you may use an LTM Self IP as per SOL13896. Ensure your APM and APM-DHCP virtual servers are in the same TMOS Traffic Group. If that is impossible set TMOS db key `tmm.sessiondb.match_ha_unit` to `false`.

(It is not mandatory to put your DHCP-related Access Policy Items into a Macro—but doing so makes the below screenshot less wide!)



An access policy consists of a start point, actions, and one or more endings. To insert a new action, click on the + sign. To configure an action or ending, click on the link inside the box. To delete an action, click on the x on the upper right edge of the box. Click the **Add Macro** button to add a purpose-built set of predefined access policy items, to simplify

Into your APM Access Policy, following your Logon Page and AD Auth (or XYZ Auth) Items (*etc.*) but *before* any (Full/Advanced/simple) Resource Assign Item which assigns the Network Access Resource (VPN), insert both Machine Info and Windows Info Items. (The Windows Info Item will not bother non-Windows clients.) Next insert a Variable Assign Item and name it “DHCP Setup”. In your “DHCP Setup” Item, set any needed DHCP parameters (explained below) as custom session variables. You *must* set `session.dhcp.servers` and place the IP address of an APM-DHCP virtual server into `session.dhcp.virtIP`. Finally, insert an iRule Event Item (name it “DHCP Req”) and set its Agent ID to `DHCP_req`.

You must attach iRule `ir-apm-policy-dhcp` to your APM virtual server.

Neither the Machine Info Item nor the Windows Info Item is mandatory. However, each gathers data which common DHCP servers want to see. By default `DHCP_req` will insert that data, if available, into requests.

See below for advanced options: DHCP protocol settings, data sent to DHCP server(s), *etc.* Typically your requests will include a user identifier from `session.dhcp.subscriber_ID` and client (machine or connection) identifiers from other parameters.

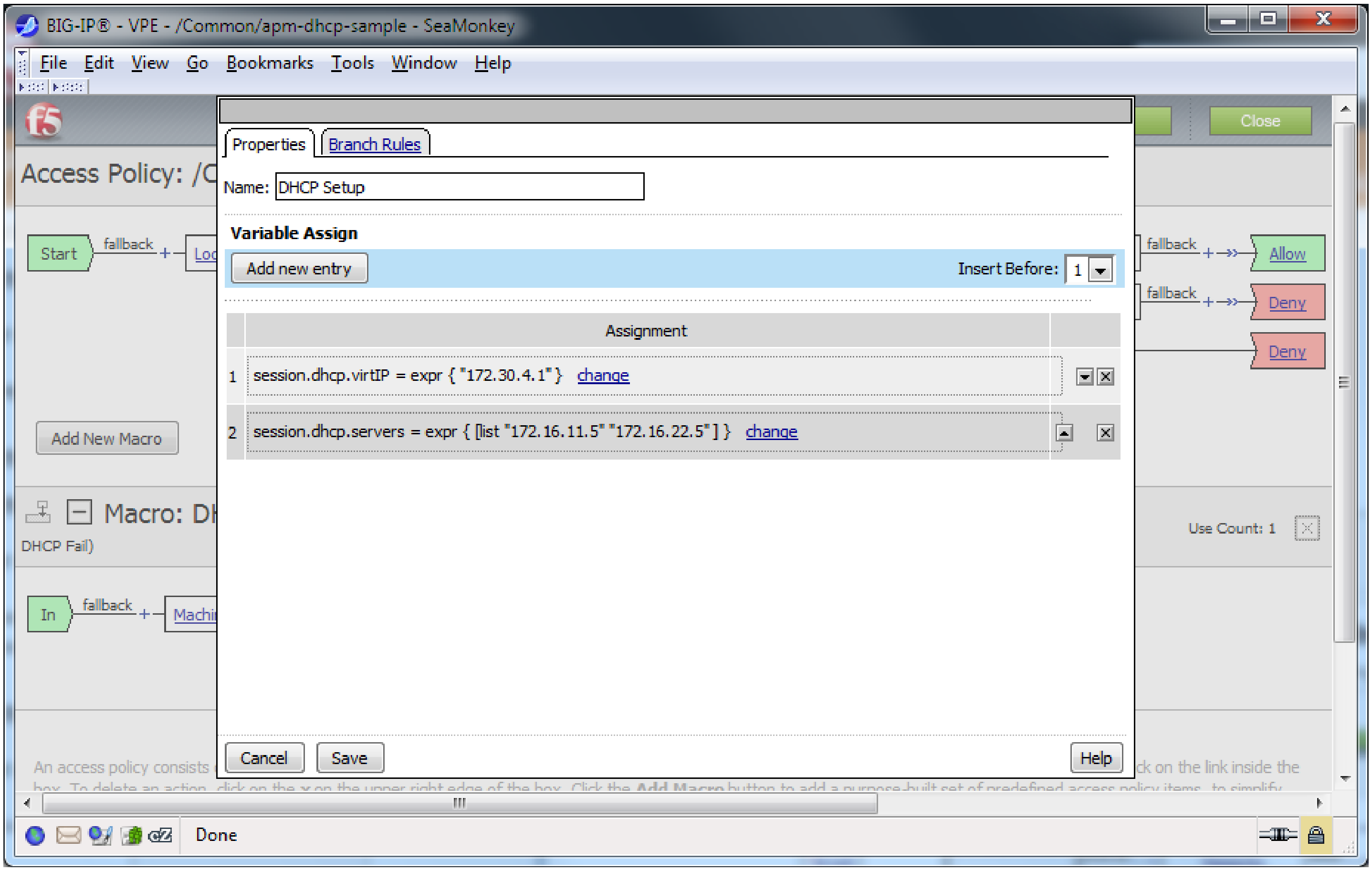
The client IP address assigned by DHCP will appear in `session.dhcp.address`. By default, the `DHCP_req` iRule Event handler will also copy that IP address into `session.requested.clientip` where the Network Access Resource will find it. You may override that behavior by setting `session.dhcp.copy2var` (see below).

Any "vendor-specific information" supplied by the DHCP server[1] (keyed by the value of `session.dhcp.vendor_class`) will appear in variables `session.dhcp.vinfo.`*`N`* where *N* is a tag number (`1-254`). You may assign meanings to tag numbers.

NB: this solution does *not* renew DHCP address leases automatically but it *does* release IP addresses obtained from DHCP after APM access sessions terminate. Please configure your DHCP server(s) for a lease time longer than your APM Maximum Session Timeout. (Since you asked: yes, this scheme supports multiple APM users and sessions in parallel.)

This solution releases DHCP address leases for terminated APM sessions every once in a while when a new connection comes in to the APM virtual server (because the BIG-IP only executes the relevant iRule on the "event" of each new connection). When traffic is sparse (say, middle of the night) there may be some delay in releasing addresses for dead sessions. You can force the release of unused leases simply by configuring a BIG-IP service monitor to connect to your APM virtual server periodically—the monitor doesn't need to log in or anything, it's just used to provoke a "connection event" to force the iRule to run.

DHCP Setup (a Variable Assign Item) will look like:



Name: DHCP Setup

Variable Assign

Insert Before: 1

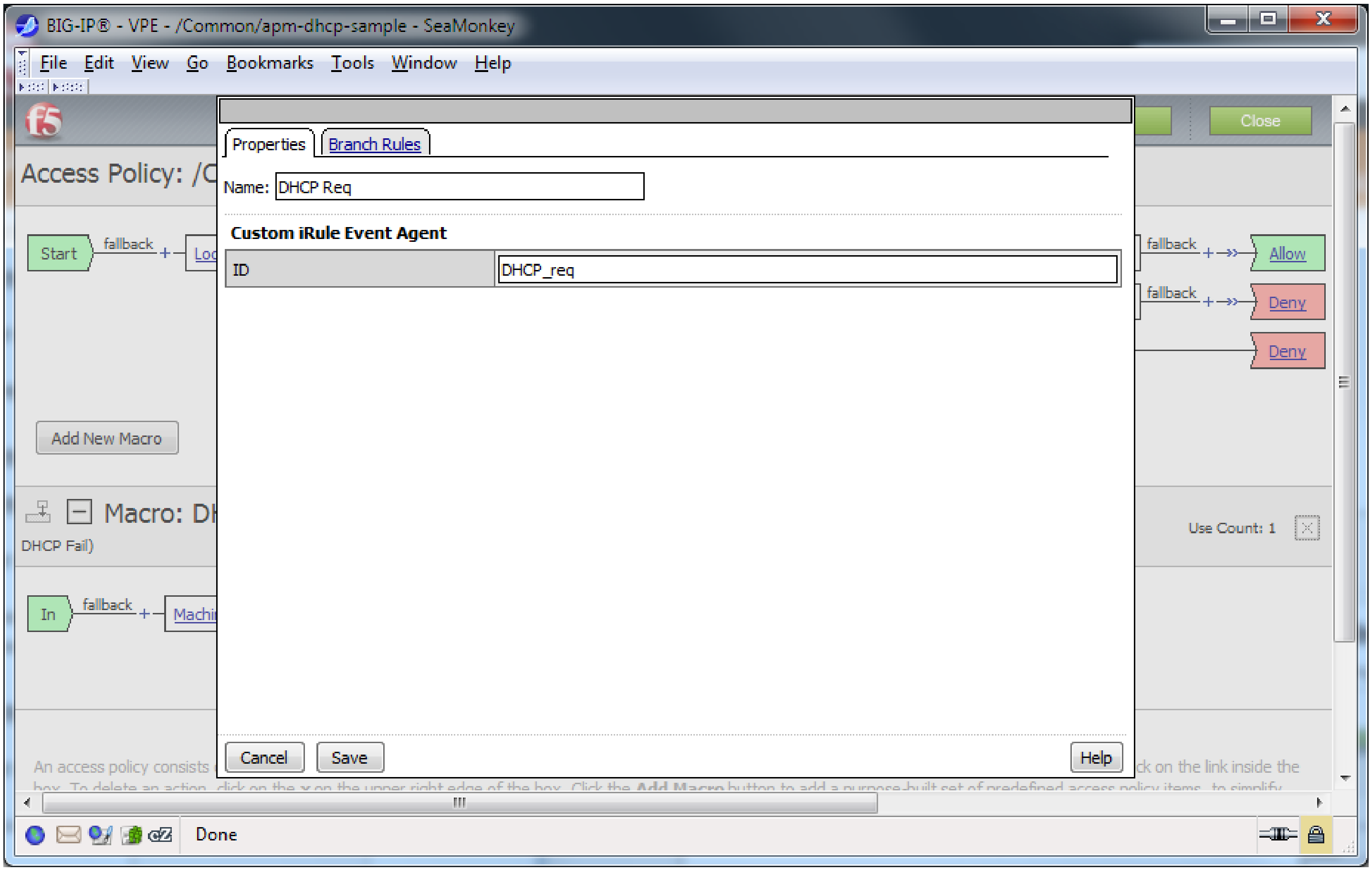
	Assignment
1	session.dhcp.virtIP = expr { "172.30.4.1" } change
2	session.dhcp.servers = expr { [list "172.16.11.5" "172.16.22.5"] } change

[1] In DHCP Option 43 (rfc2132).

Put the address of (one of) your APM-DHCP virtual server(s) in `session.dhcp.virtIP`.

Your DHCP server list may contain addresses of DHCP servers or relays. You may list a directed broadcast address (*e.g.*, “`172.16.10.255`”) instead of server addresses but that will generate extra network chatter.

DHCP Req (an iRule Event Item) will look like:

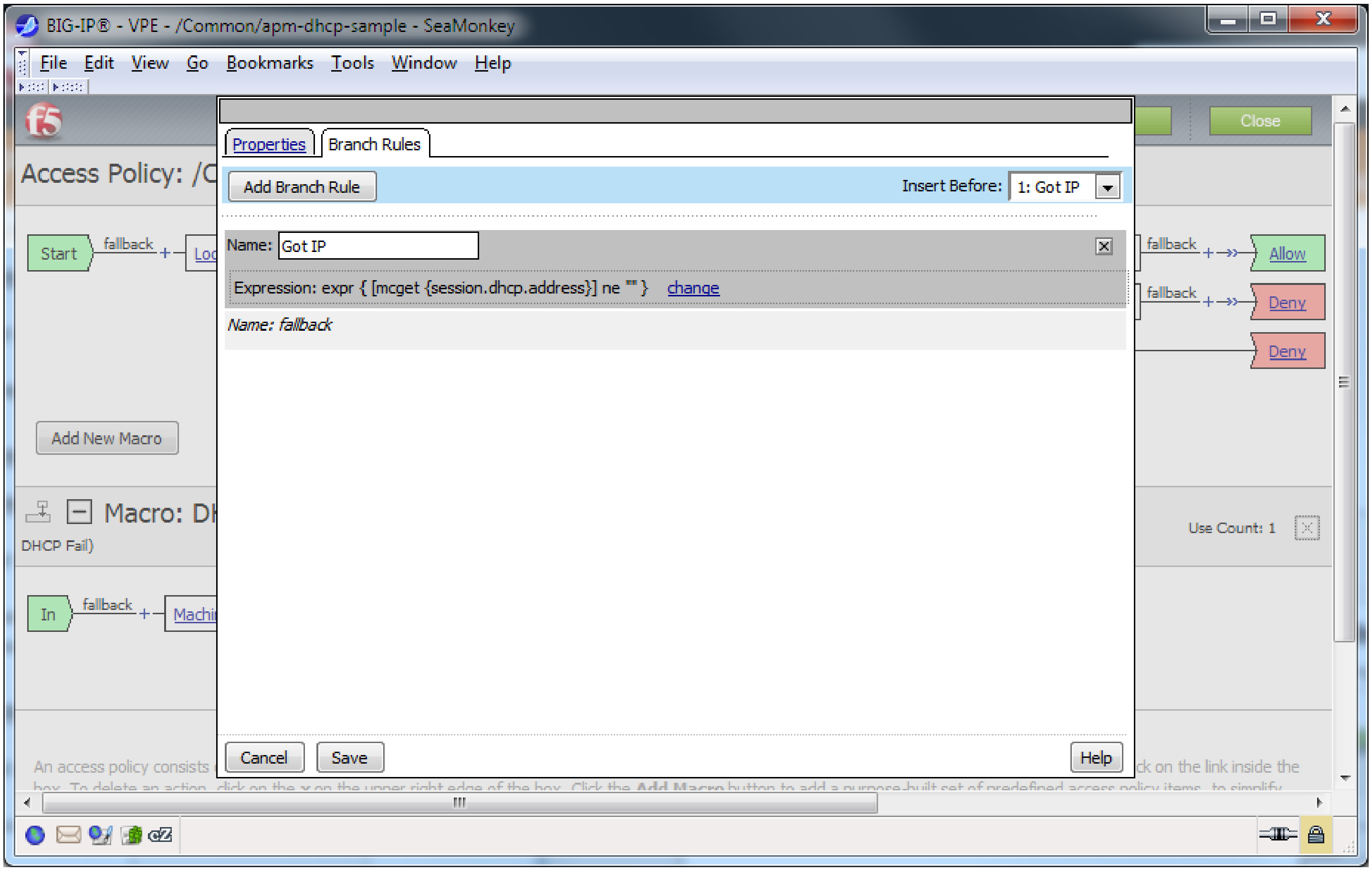


Name: DHCP Req

Custom iRule Event Agent

ID	DHCP_req

Note DHCP Req branch rules:



Insert Before: 1: Got IP

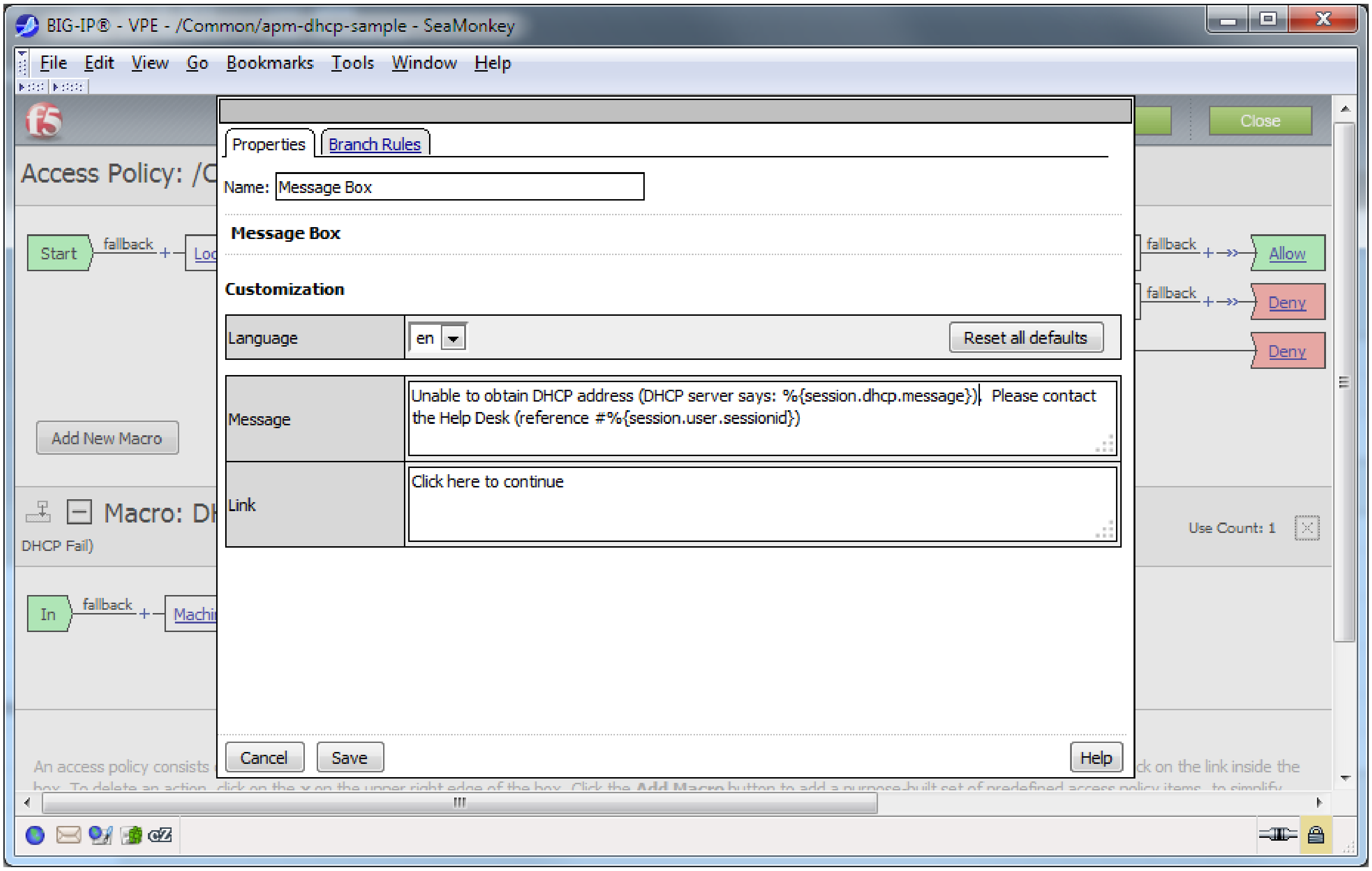
Name: Got IP

Expression: expr { [mcget {session.dhcp.address}] ne "" } change

Name: fallback

An access policy consists ... ck on the link inside the box. To delete an action, click on the x on the upper right edge of the box. Click the **Add Macro** button to add a purpose-built set of predefined access policy items, to simplify

If DHCP fails, you may wish to warn the user:


BIG-IP® - VPE - /Common/apm-dhcp-sample - SeaMonkey
File
Edit
View
Go
Bookmarks
Tools
Window
Help
Close
Access Policy: /C
Start
fallback
Loc
Add New Macro
Macro: DH
DHCP Fail)
In
fallback
Machi
fallback
Allow
fallback
Deny
Deny
Use Count: 1
Properties
Branch Rules
Message Box
Customization
Reset all defaults
Cancel
Save
Help
An access policy consists
ck on the link inside the
box. To delete an action, click on the x on the upper right edge of the box. Click the Add Macro button to add a purpose-built set of predefined access policy items, to simplify

Name: Message Box

Language: en

Message: Unable to obtain DHCP address (DHCP server says: %{session.dhcp.message}). Please contact the Help Desk (reference #%{session.user.sessionid})

Link: Click here to continue

(It is *not* mandatory to Deny access after DHCP failure—you may substitute another address into `session.requested.clientip` or let the Network Access Resource use a Lease Pool.

What is going on here?

We may send out DHCP request packets easily enough using iRules' SIDEBAND functions, but it is difficult to collect DHCP replies using SIDEBAND.[2] Instead, we must set up a distinct LTM virtual server to receive DHCP replies on UDP port 67 at a fixed address. We tell the DHCP server(s) we are a DHCP relay device so replies will come back to us directly (no broadcasting).[3] For a nice explanation of the DHCP request process see http://technet.microsoft.com/en-us/library/cc940466.aspx. (At this time we support only IPv4, though adding IPv6 would require only toil, not genius.)

By default a DHCP server will assign a client IP on the subnet where the DHCP relay device (that is, your APM-DHCP virtual server) is homed. For example, if your APM-DHCP virtual server's address were `172.30.4.2/22` the DHCP server would typically lease out a client IP on subnet `172.30.4.0`. Moreover, the DHCP server will communicate directly with the relay-device IP so appropriate routes

[2] And even more difficult using `[listen]`, for those of you in the back of the room.

[3] A bug in some versions of VMware Workstation's DHCP server makes this solution appear to fail. The broken DHCP server sends messages to DHCP relays in unicast IP packets encapsulated in broadcast MAC frames. A normal BIG-IP virtual server will not receive such packets. A proper DHCP server would not send them.

must exist and firewall rules must permit. If you expect to assign client IP's to APM tunnel endpoints on multiple subnets you may need multiple APM-DHCP virtual servers (one per subnet). Alternatively, some but not all DHCP servers[4] support the rfc3011 "subnet selection" or rfc3527 "subnet/link-selection sub-option" so you can request a client IP on a specified subnet using a single APM-DHCP virtual server (relay device) IP which is *not* homed on the target subnet but which *can* communicate easily with the DHCP server(s): see parameter `session.dhcp.subnet` below.

(The subnet(s) on which APM Network Access (VPN) tunnels are homed need not exist on any actual VLAN so long as routes to any such subnet(s) lead to your APM (BIG-IP) device. Suppose you wish to support 1000 simultaneous VPN connections and most of your corporate subnets are `/24`'s—but you don't want to set up four subnets for VPN users. You could define a virtual subnet—say, `172.30.4.0/22`–tell your DHCP server(s) to assign addresses from `172.30.4.3` thru `172.30.7.254` to clients, put an APM-DHCP virtual server on `172.30.4.2`, and so long as your Layer-3 network knows that your APM BIG-IP is the gateway to `172.30.4.0/22`, you're golden.)

When an APM Access Policy wants an IP address from DHCP, it will first set some parameters into APM session variables (especially the IP address(es) of one or more DHCP server(s)) using a Variable Assign Item, then use an iRule Event Item to invoke iRule Agent `DHCP_req` in `ir-apm-policy-dhcp`. `DHCP_req` will send `DHCPDISCOVERY` packets to the specified DHCP server(s). The DHCP server(s) will reply to those packets via the APM-DHCP virtual-server, to which iRule `ir-apm-dhcp` must be attached. That iRule will finish the 4-packet DHCP handshake to lease an IP address. `DHCP_req` handles timeouts/retransmissions, and copies the client IP address assigned by the DHCP server into APM session variables for the Access Policy to use.

We use the APM Session-ID as the DHCP transaction-ID XID and also (by default) in the value of `chaddr` to avert collisions and facilitate log tracing.

Parameters You Set In Your APM Access Policy

Required Parameters	
`session.dhcp.virtIP`	IP address of an APM-DHCP virtual-server (on UDP port 67) with iRule `ir-apm-dhcp` (no Pool or Nodes, though!). This IP must be reachable from your DHCP server(s). DHCP servers will generally assign client IP's on the same subnet as this virtual-server, though you may be able to request otherwise by setting `session.dhcp.subnet`. You may configure as many APM-DHCP virtual servers as you wish on different subnets, then set `session.dhcp.virtIP` in your Access Policy (or branch) to any one of them as a means of requesting a client IP on a particular subnet.) No default. Example (VPE syntax): `expr {"172.16.10.245"}`

[4] As of Winter 2017 the ISC, Cisco, and MS Windows Server 2016 DHCP servers support the subnet/link selection options but older Windows Server and Infoblox DHCP servers do not.

`session.dhcp.servers`	A TCL list of one or more IP addresses for DHCP server(s) (or DHCP relays, such as a nearby IP router). When requesting a client IP address, DHCP packets will be sent to every server on this list. NB: IP broadcast addresses like `10.10.10.255` may be specified but it is better to list specific servers (or relays). Default: none. Example (Access Policy syntax): `expr {[list "10.0.5.20" "10.0.7.20"]}`

Optional Parameters (including some DHCP Options)

NOTE: when you leave a parameter undefined or empty, a suitable value from the APM session environment may be substituted (see details below). The defaults produce good results in most cases. To exclude a parameter entirely set its value to `''` (two ASCII single-quotes). White-space and single-quotes are trimmed from the ends of parameter values, so `''` indicates a nil value.

It is best to put "Machine Info" and "Windows Info" Items into your Access Policy ahead of your iRule Event "DHCP_req" Item (both are safe for all clients).

`session.dhcp.firepass`	Leave this undefined or empty (or set to "false") to use APM defaults (better in nearly all cases). Set to "true" to activate "Firepass mode" which alters the default values of several other options to make DHCP messages from this Access Policy resemble messages from the old f5 Firepass product.
`session.dhcp.copy2var`	Leave this undefined or empty and by default the IP address from DHCP will be copied into the Access Policy session variable `session.requested.clientip`, thereby setting the Network Access (VPN) tunnel's inside IP address. To override the default, name another session variable here or set this to `''` to avert copying the IP address to any variable.
`session.dhcp.vendor_class` *Option 60*	A short string (32 characters max) identifying your VPN server. Default: "`f5-APM`". Based on this value the DHCP server may send data to `session.dhcp.vinfo.`*`N`* (see below).
`session.dhcp.user_class` *Option 77*	A TCL list of strings by which the DHCP server may recognize the class of the client device (*e.g.*, "`kiosk`"). Default: none (do not put `''` here). Example (VPE syntax): `expr {[list "mobile" "tablet"]}`

`session.dhcp.client_ID` *Option 61*	A unique identifier for the remote client device. Microsoft Windows DHCP servers expect a representation of the MAC address of the client's primary NIC. If left undefined or empty the primary MAC address discovered by the Access Policy Machine Info Item (if any) will be used. If no value is set and no Machine Info is available then no `client_ID` will be sent and the DHCP server will distinguish clients by APM-assigned ephemeral addresses. If you supply a `client_ID` value you may specify a MAC address, a binary string, or a text string. A value containing twelve hexadecimal digits, possibly separated by hyphens or colons into six groups of two or by periods into three groups of four, will be encoded as a MAC address. Values consisting only of hexadecimal digits, of any length other than twelve hexits, will be encoded as a binary strings. A value which contains characters other than `[0-9A-Fa-f]` and doesn't seem to be a MAC address will be encoded as a text string. You may enclose a text string in ASCII single-quotes (`'`) to avert interpretation as hex/binary (the quotes are not part of the text value). On the wire, MAC-addresses and text-strings will be prefixed by type codes `0x01` and `0x00` respectively; if you specify a binary string (in hex format) you must include any needed codes. Default: client MAC from Machine Info, otherwise none. Example: "`08-00-2b-2e-d8-5e`".
`session.dhcp.hostname` *Option 12*	A hostname for the client. If left undefined or empty, the short computer name discovered by the APM Access Policy Windows Info Item (if any) will be used.
`session.dhcp.subscriber_ID` *Sub-option 6 of Option 82*	An identifier for the VPN user. If undefined or empty, the value of APM session variable `session.logon.last.username` will be used (generally the user's `UID` or `SAMAccountName`).
`session.dhcp.circuit_ID` *Sub-option 1 of Option 82*	An identifier for the "circuit" or network endpoint to which client connected. If left undefined or empty, the IP address of the (current) APM virtual server will be used.
`session.dhcp.remote_ID` *Sub-option 2 of Option 82*	An identifier for the client's end of the connection. If left undefined or empty, the far-end IP address + port will be used.

`session.dhcp.subnet` *Option 118* *Sub-option 5 of Option 82*	The address (*e.g.*, `172.16.99.0`) of the IP subnet on which you desire a client address. With this option you may home `session.dhcp.virtIP` on another (more convenient) subnet. MS Windows Server 2016 added support for this but some other DHCP servers still lack support. Default: none.
`session.dhcp.broadcast`	Only if value is "`true`" will DHCP broadcast flag be set.
`session.dhcp.hwcode`	Controls content of BOOTP `htype`, `hlen`, and `chaddr` fields. If left undefined or empty, a per-session value optimal in most situations will be used (asserting that `chaddr`, a copy of XID, identifies a "serial line"). If your DHCP server will not accept the default, you may set this to "`MAC`" and `chaddr` will be a locally-administered Ethernet MAC (embedding XID). When neither of those work you may force any value you wish by concatenating hexadecimal digits setting the value of `htype` (2 hexits) and `chaddr` (a string of 0–32 hexits). *E.g.*, a 6-byte Ethernet address resembles "`01400c2925ea88`". *Most useful in the last case is the MAC address of `session.dhcp.virtIP` (i.e., a specific BIG-IP MAC) since broken DHCP servers may send Layer-2 packets directly to that address.*

Results of DHCP Request For Use In Access Policy

`session.dhcp.address` ← IP address assigned by DHCP !

`session.dhcp.message`

`session.dhcp.server`

`session.dhcp.expires, session.dhcp.issued`

`session.dhcp.lease, session.dhcp.rebind, session.dhcp.renew`

`session.dhcp.vinfo.`*`N`*

`session.dhcp.xid, session.dhcp.hex_client_id, session.dhcp.hwx`

If a DHCP request succeeds you will find the IP address in `session.dhcp.address`. If that is empty look in `session.dhcp.message` for an error message. The IP address of the DHCP server which issued (or refused) the IP address will be in `session.dhcp.server`. Lease expiration time is in `session.dhcp.expires`. Variables `session.dhcp.`{`lease`, `rebind`, `renew`} indicate the duration of the address lease, plus the rebind and renew times, in seconds relative to the clock value in `session.dhcp.issued` (issued time). See `session.dhcp.vinfo.`*`N`* where *`N`* is tag number for Option 43 vendor-specific information. To assist in log analysis and debugging, `session.dhcp.xid` contains the XID code used in the DHCP request. The `client_ID` value (if any) sent to the DHCP server(s) is in `session.dhcp.hex_client_id`. The DHCP request's `htype` and `chaddr` values (in hex) are concatenated in `session.dhcp.hwx`.