8

DHCP AND NETWORK ACCESS SECURITY

Security is at or near the top of every IP planner's list of networking concerns. IP address management-related security topics are no exception. There are a number of security threats to DHCP information and in its communications with those requesting information. In addition, given the role of DHCP in disseminating IP addresses for access to the network, the DHCP service itself plays a key role in providing a basic level of network access control (NAC) by virtue of its inherent function. Will you configure DHCP to provide an IP address to any device that requests one or will you configure a more discriminating policy? This chapter will delve first into the area of network access control with discussion of common strategies for deploying prudent address assignment policies. Then we'll discuss DHCP information and communications security tactics.

8.1 NETWORK ACCESS CONTROL*

The term NAC has been hyped in recent years, but the underlying concept is fundamental: identify who is attempting to access your network prior to providing such access. Various techniques are available offering various levels of access control. We'll start by

* Material in this chapter is based on Chapter 9 of Ref. 11.

IP Address Management: Principles and Practice, by Timothy Rooney

analyzing DHCP-based access control, which admittedly is among the weaker approaches to NAC. We'll then touch on more wide-reaching techniques.

8.1.1 Discriminatory Address Assignment with DHCP

Let's focus first on DHCP services and some approaches to implement discriminatory address assignment. There are several levels of policies or controls most DHCP solutions provide for discrimination of "who's asking" for an IP address via DHCP. The first is to simply filter requests by an available form of client identifier such as the MAC address of the client requesting an address. Recall that the MAC address is found in the client hardware address (chaddr) field of a DHCPv4 packet. DHCPv6 device identifiers consist of the Device Unique ID (DUID) and identity associations (IAs) that identify each client and interface, respectively.

If the DHCP server has a list of acceptable (and/or unacceptable) device identifiers, it can be configured to provide a certain IP address and associated parameters to those clients with an acceptable identifier, and either no IP address or a limited function IP address to those without an acceptable device identifier. By *limited function IP address*, we mean that the network routing infrastructure is preconfigured to route IP packets with such source IP addresses to only certain networks, such as to the Internet only, or even nowhere. An IP packet with source address A may be routable across the enterprise, while one with source address B may be routable only to the Internet, for example.

This type of IP address and configuration assignment is also possible by filtering on the client class of the device requesting an IP address, as we discussed in Chapter 4. Certain clients, such as VoIP phones, provide additional information about themselves when requesting an IP address in the vendor class identifier field of the DHCP packet. The user class identifier field may also be used. The DHCP server can be configured to recognize user classes and/or vendor classes of devices on your network to provide additional information to the DHCP server when requesting IP address and configuration parameters. Addresses can be assigned from a certain pool and/or additional configuration parameters can be assigned to the client via standard or vendor-specific DHCP options.

Another level of discriminating IP address assignment is possible by authenticating the user of the machine requesting an IP address. This function can be used in conjunction with device identifier and client class discrimination described above. For example, if a client with an unknown or unacceptable device identifier attempts to obtain an IP address, one option is to completely deny an address; another option is to require the user of the client to login via a secure access web portal page.

This enables easier capture of new device identifiers for legitimate users of your network. (Those users sometimes pop in new interface cards!) Solutions ranging from perl scripts such as NetReg (90) to sophisticated integrated software solutions are available to direct such users to a login/password requesting web page. A simple lookup against a database of legitimate users then allows access or denial of the client to a production IP address. These systems typically work in accordance with the packet flow shown in Figure 8.1.*

* DHCPv4 process is shown, but a comparable flow could be employed with DHCPv6.

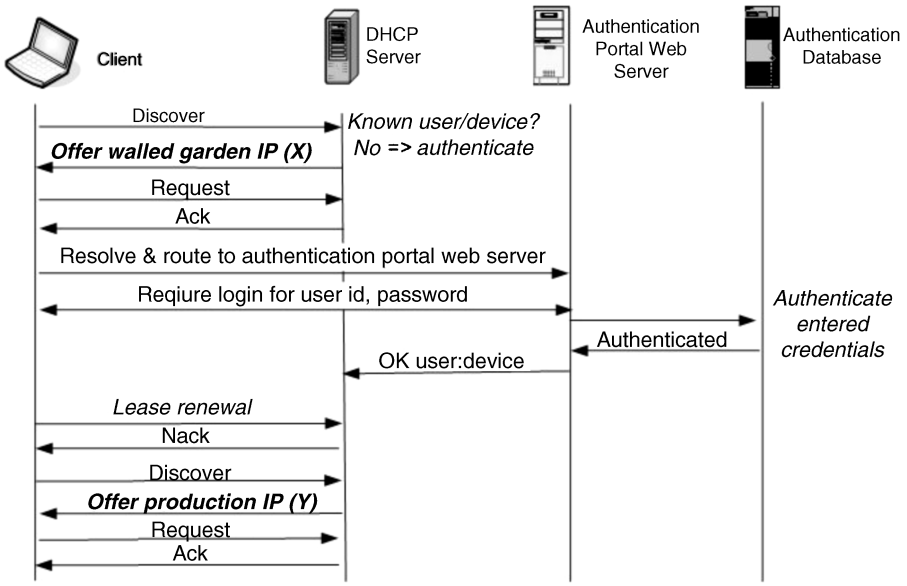

Figure 8.1. Basic DHCP captive portal flow (11).

Walking through this flow, the process begins with a device connecting to the network, attempting to obtain an IP address via DHCP. The DHCP server, employing device identifier or client class-type filtering discussed above, determines if the device is a known user device.* If the device is known or is otherwise already authenticated, the DHCP process may continue with an Offer for a production IP address, followed by a Request and Ack. However, if the device is not known or is required to be authenticated, the DHCP server can still provide an IP address by completing the DORA process; but the IP address assigned in this case would be a *captive portal*, *walled garden,* or *quarantined* IP address.

These terms refer to the fact that the IP address assigned to the client will only be routed to the subnet or VLAN that has the authentication web server and associated servers running. This quarantined VLAN enables IP communications but only to this restricted set of devices. This cordons off the device from infiltrating the rest of the network until the corresponding user can be authenticated. The routing infrastructure must be configured to route packets with a source address from the quarantined address pool to the quarantined VLAN and/or the client must be configured with the classless static route option. Thus, address X as shown in Figure 8.1 is a member of the quarantined VLAN, on which only limited network resources are available. Figure 8.2 illustrates an example network topology of this captive portal configuration.

Now when the user opens up a web browser, he/she can type in any web address. A *limited configuration* DNS server is required on the quarantined VLAN, limited in the sense that it will resolve any and every query to the IP address of the authentication web

* In some cases, even known user devices may require periodic reauthentication as a security precaution.

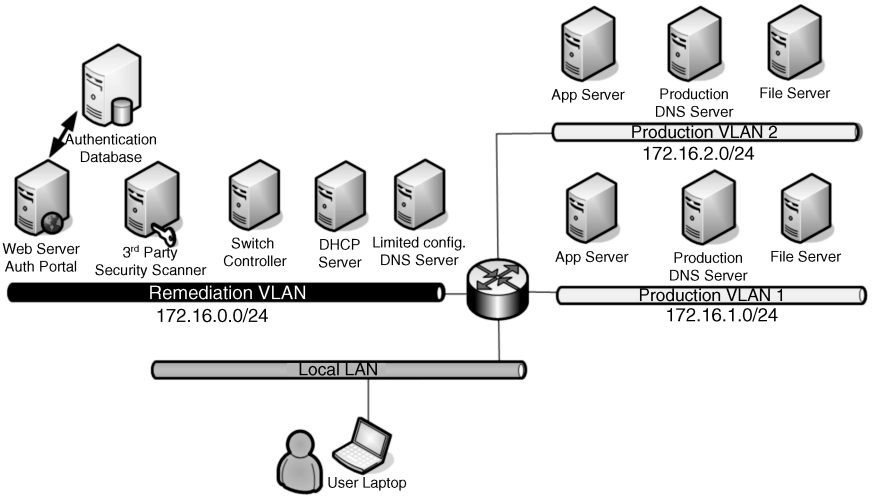

Figure 8.2. Captive portal network diagram (11).

server. Thus, no matter what web address is entered in the web browser, the web address is resolved by the captive portal web server. The authentication web server presents the login page. You may have seen something similar to this if you travel and use a hotel's broadband or wireless service. Once the requested credentials are entered, which for an enterprise environment would typically comprise a user ID and a password, the web page can call a CGI script to pass the entered credentials to a back-end database. This authentication database could be an LDAP server, a Windows Domain Controller, a Radius server, or other form of authentication data store.

Based upon the results of the authentication, the requesting device would then be deemed authorized or not, and if authorized, optionally what class of authorization is granted. The class of authorization provides more granularity than a simple boolean "authorized or not," where different authorized users can be assigned a different production IP address, which in turn can provide access to different network resources. For example, basic level users may be granted access to a basic set of resources, while advanced level users may be granted access to additional resources, for example, IT resources. Once again this requires the routing topology be configured with multiple source-routed or VLAN segments, with these networks and corresponding routing plan mapped to DHCP server configurations in terms of associating address pools with service levels.

The manner in which the production IP address is assigned follows from expiration or denial of renewal of the quarantined IP address. The quarantined IP address lease time is generally configured as a short lease time ($\sim$1–5 min). This enables the device to attempt to renew quickly. Should the device still be in the process of authentication, its renewal attempt would be ACK'd, extending the lease. Once authentication is completed successfully, the authentication system updates the DHCP server to add the client MAC

address to the "known" or "allow" pool. The renewal attempt for the quarantined address would then be NAK'd, enabling a fresh DORA process to provide a "production" IP address (address Y in Figure 8.1). Should the device fail authentication, the renewal can be NAK'd and subsequent address attempts denied; alternatively, the quarantined address renewal attempt can be granted in order to provide access only to resources on the quarantined network if desired.

Beyond these device and user identification measures based on device identifiers, client classes, and user authentication, this general flow can also provide additional validation on the machine requesting the IP address. The DHCP process can be used to invoke an external security scanning system like Nessus, or another third-party application to scan the requesting client for viruses, or to validate the use of acceptable virus protection software. This device-scanning step can be used alone or in conjunction with the device identification measures to provide a robust access security solution via DHCP.

One example network configuration for DHCP-based secure access is depicted in Figure 8.2. The DHCP server shown in the diagram would be configured with a number of client class sets. We refer to the client class as the matching criteria in the DHCP packet, and link this to client class set mapping to the associated network accessibility. For example, we would need a client class set for at least each of the following in our example:

- Captive portal network (remediation VLAN)
- Production network 1
- Production network 2

Think of these client class sets as bins into which individual clients are placed based on the linking of their authentication state to the device's client class. Thus, client class members would be categorized by the DHCP server in accordance with defined client classes as they appear on the network and users authenticate. These client classes would generally map to pool definitions on the DHCP server as shown in the following simple example ISC server configuration (35). Note that additional options can be defined for each of the pools to provide additional configuration granularity to clients falling into each set or pool.

```
subnet 172.16.0.0 netmask 255.255.252.0 {
# subnet level options here...
  pool{                                          #captive portal pool
    range 172.16.0.10 172.16.0.254;
    option domain-name-servers 172.16.0.5;       #limited config DNS server
    default-lease-time 150;                      #short lease time
    allow unknown clients;                       #clients not predefined.
  }
pool {                                           #Prod Net 1
    range 172.16.1.10 172.16.1.254;
    option domain-name-servers
```

```
     172.16.1.5;                              #production DNS server
    default-lease-time 14400;                 #normal lease time
    deny unknown clients;                  #clients must be predefined.
    allow members of "net1";               #client class net1 allowed
  }
pool {                                     #Prod Net 2
    range 172.16.2.10 172.16.2.254;
    option domain-name-servers
     172.16.2.5;                           #production DNS server
    default-lease-time 14400;              #normal lease time
    deny unknown clients;                  #clients must be predefined.
    allow members of "net2";               #client class net2 allowed
  }
}
```

Based on the results of the authentication process, the authentication server must be able to update the DHCP configuration to place the client into the appropriate bin or class. Thus, if the device is successfully authenticated for access to production network 2, the authentication portal needs to add the specific device's client class value (e.g., MAC address) to the client class group for production network 2 ("net2" class in the example above). This update may be performed, for example, using the ISC DHCP server OMAPI interface (requires version 3.1 or above). This client class declaration can define class-specific options on the DHCP server to provide to the client, for example, default gateway, DNS server, along with any other option.

The captive portal VLAN may only consist of "unknown clients," a designation configurable with the ISC DHCP server. The captive portal network (the remediation VLAN in the figure) is deployed, including the limited configuration DNS server, web server as the authentication portal, with access to an authentication database, and optionally a security scanning server and any other required preaccess services.

More than one DHCP server may be deployed for high availability and/or for scaling for larger networks. This approach does complicate things, as the DHCP server configurations need to be consistent on both servers to route unknown clients or clients requiring authentication to the captive portal net.

8.2 ALTERNATIVE ACCESS CONTROL APPROACHES

You may be thinking that the DHCP-based approach is fine for clients utilizing DHCP; but what about those "clever users" who figure out the subnet address and then manually encode a static IP address on their machines to access the network? These clever users may after all be those of most concern from a secure access perspective. In addition, for devices using IPv6 stateless autoconfiguration, no DHCP interaction is required for address assignment.

There are three basic alternative approaches for enabling detection and associated remediation action of devices without relying on the DHCP-based approach. We'll talk about leading networking vendor NAC approaches in the next section.

- DHCP LeaseQuery
- Layer 2 switch alerting
- 802.1X

8.2.1 DHCP LeaseQuery

If most or all addresses on a subnet are configured using DHCP by policy, that is, each IP address *should* have a corresponding DHCP lease, then the LeaseQuery approach may be used. The DHCP LeaseQuery is a DHCP protocol message that enables an edge router to query the DHCP server regarding the lease status of a particular device or set of devices. This provides some assurance that a device attempting to communicate via the router has not spoofed an address that should have been assigned by the DHCP server.

When the router receives IP traffic within a layer 2 frame from a particular MAC address, for example, it can issue a DHCP LeaseQuery message to its configured DHCP servers (i.e., in its role as relay agent). If a DHCP server had previously provided a lease for the client, it will respond to the router, and the router will give the green light and route the device's packets. If not, the device does not have a lease and the router can drop the device's packets. The router can cache this information as well so that the LeaseQuery rate is not exceedingly high. Of course, this form of access control applies only when all clients on a subnet use DHCP such as in broadband access networks, not when other statically addressed devices communicate on the subnet.

8.2.2 Layer 2 Switch Alerting

Another approach takes advantage of SNMP-enabled switches to issue an SNMP trap upon a link up state on one of its ports and to accept port-level VLAN configurations. This alerting capability along with SNMP writable configuration information can enable gatekeeper-like functionality by dynamically identifying devices attempting to access the network and configuring the switch to provision the port to a particular VLAN. A third-party system or product would be needed to process traps, make decisions on appropriate VLAN assignments, and configure the switch accordingly.

Let's look at how this would work. If we consider the process of a device connecting to a network from the beginning, the device "boots up" on the network from layer 1. Thus, the physical layer/electrical connectivity is first attained; then the data link layer is initialized whereby layer 2 frame synchronization occurs. Then layer 3 follows, with the issuing of a DHCP packet, for example, or directly issuing IP packets if a static address is configured at layer 3. As the data link layer initializes (prior to layer 3), the switch to which the device is connected will deem the "link up" and issue a trap. Because the trap is sent prior to layer 3 initialization, this scheme can identify both statically addressed and DHCP-addressed devices.

Traps would be directed to a system that can identify the link up state, ascertain the link layer (MAC) address of the newly connected device, and then determine whether the device requires authentication or validation. This determination can be made via a MAC address database within the system that identifies known or acceptable MAC addresses

and differentiates these from unknown or known unacceptable MAC addresses. The system would associate these two or perhaps more MAC address categorizations with corresponding VLAN assignments, which would then be programmed on the corresponding switch for the given port. The connected device would then be connected to the assigned VLAN. You can probably see the analogy to the DHCP scenario we discussed using client classes. In this case, the third-party system uses its database and configures the layer 2 switch using SNMP or other means instead of assigning an IP address using DHCP.

For quarantined or captive portal access, the VLAN assignment would lead only to the authentication network. For those passing authentication and/or device validation, the system could reassign the MAC address to the acceptable list and then configure the switch accordingly to change the ports VLAN association. Depending on the authentication method, client software may or may not be required. For web-based login/password, it may not be necessary to configure each of your client computers with authentication clients. However, if Radius, or other challenge/response authentication strategies are employed, client software will be necessary.

8.2.3 802.1X

IEEE 802.1X is a protocol specification enabling edge device capture of new access attempts, with the use of Radius authentication and dynamic switch port configuration. You may have used Radius in the days of prebroadband Internet dial up, which used the Point-to-Point protocol at layer 3. 802.1X, developed by the IEEE 802.1 working group focused on layer 2 protocols, is as you'd expect, a layer 2 protocol. Like the switch-based authentication strategy discussed in the previous section, this approach operates at layer 2, prior to the device getting a layer 3 (IP) address via DHCP. And 802.1X is based on standards, which theoretically enables the use of different vendors' products as components within the overall solution.

As depicted in Figure 8.3, 802.1X requires a client or agent called a *supplicant*, which interacts with an *authentication server* by way of an *authenticator* (e.g., switch). Upon initial connection to a network, the supplicant utilizes the Extensible Authentication Protocol (EAP) over 802.1X to initiate a connection request to the network access device. The switch can be configured to block all traffic by default except EAP packets from unauthenticated ports.

The access switch to which the device is connected at the data link layer transmits the EAP traffic to the authentication (i.e., Radius) server. The Radius server, in turn, challenges the client for an ID and password. Upon successful authentication, the Radius server communicates with the edge device to enable access to the associated device's port.

8.2.4 Cisco Network Admission Control

Cisco's Network Admission Control (NAC) offering (91) is based primarily on 802.1X. It requires a Cisco Trust Agent (CTA) optionally with a Cisco Security Agent installed on each end user device (Figure 8.4). The Trust Agent contains a Radius client. Upon initial

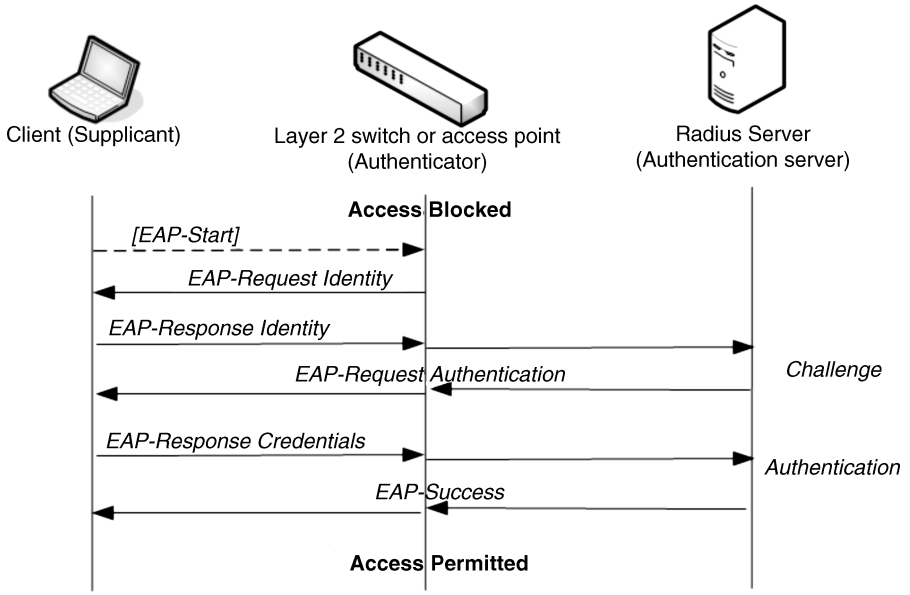

Figure 8.3. 802.1X authentication.

connection to a network, the CTA utilizes the Extensible Authentication Protocol over 802.1X or UDP to initiate a connection request to the network.

The network access device, typically a switch to which the device is connected at the data link layer, transmits the EAP traffic to the Cisco access control server (ACS), which provides Radius services. This Radius component, in turn, challenges the client for an ID and a password. A third-party validation solution may be invoked to scan the device

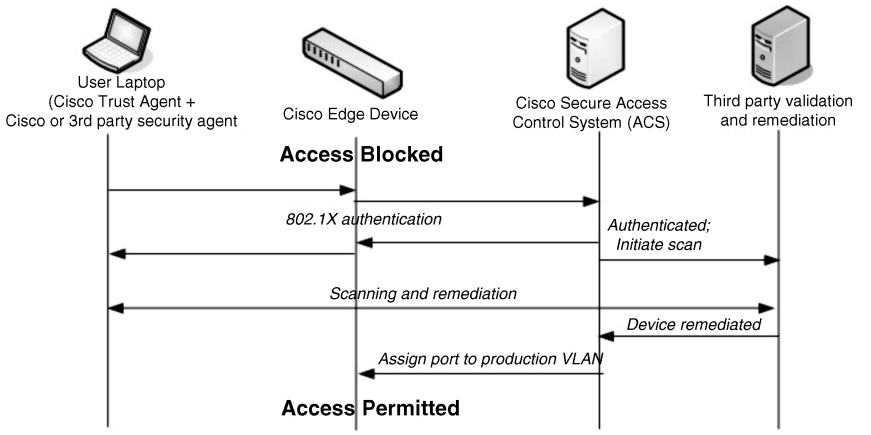

Figure 8.4. Cisco NAC basic flow.

attempting to gain access. Upon successful authentication and validation, the ACS communicates with the edge device to enable access to the associated device's port.

8.2.5 Microsoft Network Access Protection

Microsoft introduced Network Access Protection (NAP) (92) to enable administrators to ensure computers accessing a network have appropriate software installed at or above a specified version and is supported in Microsoft Vista™ or 7 client as well as Windows Server 2008 implementations. Microsoft supports an API to enable other vendors to support NAP technologies. NAP primarily emphasizes device compliance and health, and as a by-product, access control. That is, NAP is intended to enable network administrators to validate device compliance with current software releases upon network access, and not inherently to prevent access by malicious attackers. Nonetheless, NAP does entail denial of access pending health status verification with its three key functions:

- *Health Policy Validation.* Upon a network access attempt, a device's "health state" is obtained and compared with the administrator's defined "health policies." If the device complies with the specified health policies, the device is permitted unrestricted access; if the device is not compliant, the device can be provided restricted access or full access if in monitoring-only mode.
- *Health Policy Compliance.* Noncompliant devices can optionally be automatically upgraded upon access attempt. If in monitoring-only mode, the device will enjoy full network access. In restricted access mode, the device will have only restricted network access until compliance is achieved.
- *Limited Access.* Administrators can constrain the scope of network accessibility to noncompliant devices.

Microsoft Vista clients contain a NAP client that communicates with a NAP Policy Server (NPS), part of Microsoft Windows Server 2008, during an attempt to access a network. The NPS enforces the compliance policies and is consulted during access attempts over a variety of technologies, including IPSec, 802.1X, VPN, Radius, and DHCP. IPSec is the strongest form of policy enforcement and consists of a Health Registration Authority (HRA), which issues X.509 certificates to compliant NAP Enforcement Clients (ECs) based on compliance verification by the NPS.

The 802.1X access flow follows that described above for 802.1X, with the addition of the NAP Policy Server validating the device's compliance. VPN, Radius, and DHCP access components include a NAP Enforcement Server (ES) and a NAP EC that communicate regarding policy compliance during access attempts to the network via the respective technologies (Figure 8.5).

The client device, or NAP client, contains a NAP Agent, as well as Microsoft-provided and API-accessible System Health Agents (SHAs) and NAP ECs in support of additional applications. Likewise, the NPS features an API for corresponding System Health Validators (SHVs). When attempting to access the network, the NAP client will provide a Statement of Health (SoH) to the NPS via the corresponding access server; for

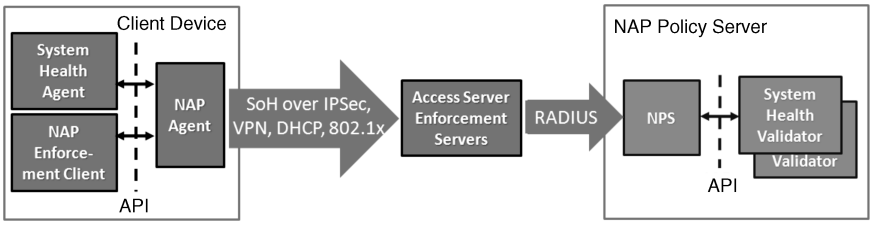

Figure 8.5. Microsoft NAP components (92).

example, the HRA, VPN server, DHCP server, and others. The access server passes this on to the NPS, which validates the policy compliance and either permits or restricts access based on the access technology and NAP policies. Full or restricted access is conveyed to the NAP EC on the client for enforcement, though other network configuration such as router access lists or static routes may also be required.

8.3 SECURING DHCP

8.3.1 DHCP Threats

Within enterprise environments, most threats to DHCP are posed by internal (i.e., intraorganizational) clients. DHCP servers should not be reachable by external clients by simply not deploying DHCP servers on external subnets nor relaying DHCP packets from external sources. For service providers that initialize subscriber devices using DHCP, whether cell phones, cable or fiber routers, and so on, threats to DHCP service can by definition originate externally to the network. In short, all organizations using DHCP are vulnerable. The degree of vulnerability and the impacts of compromise should drive the response in the form of securing DHCP to minimize such impacts. We'll look at the major forms of attack next.

Like all network services, DHCP is vulnerable to denial of service (DOS) attacks. When an attacker floods a given server with requests too numerous for the server to handle, the server may spend all its cycles attempting to deal with the flood and not on legitimate client requests; thus, these legitimate clients go unserved, and service is denied to them.

Another type of attack involves a rogue client attempting to obtain a valid IP address and configuration to access the network. This could be malicious, for example, theft of broadband service, or merely accidental, for example, a visitor plugging into the wall jack in the conference room.

A third form of attack features a rogue DHCP server that responds to lease requests from clients with incorrect IP address and/or option parameter information. This "man in the middle" type of attack may attempt to set improper configuration parameters on the client, such as the default gateway or DNS server address(es) to use. Note that with IPv4, a rogue DHCP server attack is generally only applicable when the server is on the same subnet as the client; relay agents presumably would be configured to relay DHCP packets

to authorized DHCP servers. A remote rogue DHCPv6 server may be reachable via the DHCP multicast address.

The client may receive DHCPOFFERs from both the legitimate DHCP server(s) and the rogue server. Many clients will select the first offer that includes its requested parameters. If the rogue server is on the same subnet as the client, and legitimate servers are not, then it's likely the rogue server may be able to specify the IP configuration of the client.

8.3.2 DHCP Threat Mitigation

Protection against DOS attacks should be implemented in a broader context beyond just DHCP. Other potential targets within an organization, including DNS servers or web servers, imply that a gateway-based or packet filtering approach be considered to protect all servers with a common solution. Such a solution typically involves packet filtering and threshold limiting of the number of outstanding packets in process, though care must be taken with DHCP since most clients' transactions are funneled through DHCP Relay agents, concentrating packets from a given set of source addresses.

Mitigation steps for the threat of unknown clients accessing the IP network by illicitly obtaining an IP address from DHCP requires identification of clients based on various access control techniques we discussed at the beginning of this chapter.

Rogue DHCP servers may be difficult to detect, especially for clients on the same subnet as the rogue server. But both ISC and Microsoft implementations provide means to mitigate rogue servers. For ISC, use the `authoritative` directive, which configures the server to issue a DHCPNAK if a client requests a lease for an address for which the server is authoritative yet for which the server has no record. Microsoft requires DHCP servers to be authorized within Active Directory; thus, when a Windows DHCP server boots, it verifies its authorization in Active Directory before processing DHCP packets.

8.3.3 DHCP Authentication

The IETF has defined DHCP authentication in RFC 3118 (47) as a mechanism providing validation of the sender and receiver of DHCP packets via the use of shared tokens or keys. A token is simply a fixed value that is inserted into the DHCP Authentication option field. The receiver of the packet examines the token and if the token matches its configured token, the packet is accepted; otherwise, it is dropped. This method provides weak endpoint authentication and no message verification. The use of shared keys can provide stronger endpoint authentication with message verification. However, shared keys must be configured on each client, with each client's key configured on each DHCP server through which the client obtains leases. The DHCP Authentication specification does not define the mechanism for key distribution. Mobile clients, for example, would need to be configured with tokens for each DHCP server with which they may interact and vice versa.

Here's how DHCP authentication works. The client creates an HMAC-MD5 hash of its DHCPDISCOVER packet and signs it using the shared key. The resulting digest is

placed in the DHCP Authentication option and transmitted within the DHCPDISCOVER packet to the server. For the purposes of the hash computation, the hash portion of the DHCP Authentication option must be set to zero. The DHCP server would then compute a hash of the received message utilizing the shared key associated with the client (identified by the secret ID field of the DHCP Authentication option). The server zeroes out the hash value, hops, and GIAddr fields for the purposes of the hash computation. If the calculated hash matches that transmitted in the original DHCP Authentication option, the client and the contents of the packet are considered authenticated. The DHCP server utilizes the same shared key to compute the hash value of its DHCP Authentication option when it prepares its DHCPOFFER and future packets to the client.

There have been very few implementations of DHCP Authentication. The challenges of key management and processing delays due to hash computation have been deemed too heavy a price to pay for the perceived benefits. Security of the DHCP service then typically falls on DHCP server administrators to monitor servers and react to threats as they occur.