Guide to the Secure Configuration and Administration of iPlanet Web Server, Enterprise Edition 4.1 ®

The Network Applications Team

of the

Systems and Network Attack Center (SNAC)

Written by:
James M Hayes, Capt, USAF

National Security Agency
ATTN: C43 (Hayes)
9800 Savage Rd.
Ft. Meade, MD 20755-6704

410-854-6191 Commercial
410-854-6510 Fax
jimh@thematrix.ncsc.mil

Distribution is limited to U.S. Government
Entities and their contractors

Dated: January 3, 2001
Version 1.0

This page intentionally left blank.

Warning

Caution: You can severely impair or disable a Windows NT System or iPlanet Web Server with incorrect changes or accidental deletions when using a registry editor (Regedt32.exe or Regedit.exe) to change the system configuration.

Currently, there is no “undo” command for deletions within the registry. Registry editor prompts you to confirm the deletions if “Confirm on Delete” is selected from the options menu. When you delete a key, the message does not include the name of the key you are deleting. Therefore, check your selection carefully before proceeding.

Trademark Information

iPlanet Web Server, Enterprise Edition and iPlanet Web Server Administration Server, and iPlanet Directory Server are registered trademarks of the Sun-Netscape Alliance in the U.S.A. and other countries.

Netscape Communicator and Netscape Navigator are registered trademarks of Netscape Communications Corporation in the U.S.A and other countries.

Windows NT and Windows Notepad are registered trademarks of Microsoft Corporation in the U.S.A. and other countries.

About the Guide to the Secure Configuration and Administration of iPlanet Web Server, Enterprise Edition 4.1 SP4

The iPlanet Web Server, Enterprise Edition 4.1 SP4 is produced by the Sun-Netscape Alliance™. It is the replacement for the Netscape Enterprise Server. This document describes how to securely install, configure, and administer the iPlanet Web Server, Enterprise Edition Server and iPlanet Web Server Administration Server. It is intended for the reader who is already familiar with the iPlanet servers but needs to understand how to install, configure, and administer the product in a secure manner. The information presented here is written in a direct and concise manner in deference to this intended audience – very little introductory material is provided. The remainder of this document will refer to the iPlanet Web Server, Enterprise Edition as the iPlanet Web Server (iWS) and the iPlanet Web Server Administration Server as the Administration Server.

This document should be used in conjunction with The *"Guide to Securing Microsoft Windows NT Networks"* to provide a more secure implementation of the iPlanet servers. Some iPlanet security issues, and corresponding configuration and administrative actions, are very specific to the way the product is being used. For this reason, it is difficult in some areas to recommend specific, concrete actions. Instead, alternatives are offered which describe the concerns and recommend solutions based on specific requirements.

PLEASE NOTE THAT THIS DOCUMENT ASSUMES THAT THE READER IS A KNOWLEDGEABLE WINDOWS NT ADMINISTRATOR. A knowledgeable Windows NT administrator is defined as someone who can create and manage accounts and groups, understands how Windows NT performs access control, understands how to set account policies and user rights, is familiar with how to setup auditing and read audit logs, etc. This document does not provide step-by-step instructions on how to perform these basic Windows NT administrative functions – it is assumed that the reader is capable of implementing basic instructions regarding Windows NT administration without the need for detailed instructions.

PLEASE NOTE THAT THIS DOCUMENT ASSUMES THAT THE READER IS KNOWLEDGEABLE ABOUT LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL DIRECTORY ADMINISTRATION. A knowledgeable directory administrator is defined as someone who can create and manage accounts and groups, implement account policies and user rights, is familiar with how to setup auditing and read audit logs, load certificates into a directory, etc. This document does not provide step-by-step instructions on how to perform these functions. In addition, it is assumed that the reader has some familiarity with public key infrastructure (PKI) - specifically certificate authorities and X.509 v3 certificates.

This document consists of the following chapters:

Chapter 1, “iPlanet Web Server, Enterprise Edition Installation,” provides an overview of the pertinent security issues related to the installation of the Enterprise and Administration Servers.

Chapter 2, “iPlanet Web Server Administration Server,” describes the security issues related to the administration of iWS and Administration Server.

Chapter 3, “iPlanet Web Server, Enterprise Edition,” describes how to securely configure iWS features.

Chapter 4, “Final Thoughts,” takes a quick look at the importance of antiviral programs.

Appendix A, “References,” provides reference information for sources used in this document.

Appendix B, “iWS Problems of Interest,” provides reference information for known security and operational issues of iWS.

Table of Contents

An Important Note About Operating System Security

iWS and Administration Server's security are tightly coupled to the operating system. Therefore, it is critical that Windows NT Server be securely implemented prior to installing and configuring iWS and Administration Server. Equally important to the overall security of the system are file permissions, registry settings, password usage, user rights, and other issues associated with Windows NT security.

The recommended source of information for how to securely configure the Windows NT 4.0 Server is the "*Guide to Securing Microsoft Windows NT Networks*." This document is made available as a service to U.S. Government entities and their contractors with various recommended settings for the system registry. Administrators, when possible, are advised to implement security guidelines on a test LAN prior to implementation on a production server – we can not guarantee that problems will not arise as our testing can not possible cover all possible configurations.

This page intentionally left blank.

Chapter

1

iPlanet Web Server, Enterprise Edition Installation

Pre-Installation

There are a number of security related actions that must be performed prior to the installation of the iWS.

Windows NT 4.0 Security Recommendations

Prior to installing iWS on Windows NT 4.0 Server, the operating system should be secured based on the "*Guide to Securing Microsoft Windows NT Networks*" developed by the Systems and Network Attack Center (SNAC) of C44. This guide includes extensive detail about securing the Windows NT operating system, applying Service Pack 6a and Hotfixes, and many other important Windows NT security issues. These preliminary steps are critical to ensuring the most secure configuration of iWS.

Restricted User Account

Prior to iWS installation, a user account should be created for running iWS. This user should be a member of the Windows NT Users Group and be granted the "Log on as a service" and "Log on locally" rights. This account will be assigned as the owner of the iWS service when it starts up as described in the Post-Installation section. Use of this restricted account will limit the damage that might occur should the iWS be compromised. Note that this user account can only be used on iWSs that do not use SSL. On installations that use SSL, the iWS server must be run under the system account in order for it to function properly.

In order to function properly, the Administration Server must run using the System Account. If the restricted user account is used, some administration functions will not work, i.e., can not turn a web server instance off via the Administration Server.

iPlanet Administrator Group

In addition to the iWS user account, a Windows NT iPlanet Administrators Group should be created for users who will be assigned special iWS administration rights.

Installation

Installation of iPlanet Web Server on a Separate NTFS Partition

For best security, iWS should only be installed on a server and not on a NT domain controller. In addition, iWS and Windows NT should be installed on separate NTFS partitions; however, do not install iWS on an NFS-mounted drive due to potential security and file locking restrictions on remote partitions. NTFS is recommended as it supports the enforcement of access controls over directories and files necessary for the secure operation of the Administration Server and iWS.

Administration Server Superuser Account

During installation of iWS, the Administration Server Superuser account is created. This account is strictly an iPlanet account and does not exist on the Windows NT system. By default, the login name is admin and the password is specified at the time of installation. The login name should be changed to something less obvious and the password should be based on the recommendations from "*Guide to Securing Microsoft Windows NT Networks.*" In addition, this password should not be the same as the NT Administrator account's password.

Administration Server Port Selection

During installation, the Administration Server port is configured. The default port setting is 8888. A random port number should be chosen as a simple countermeasure that will somewhat mask the location of the port. Any unused port above 1024 should be used.

Administration Server and iPlanet Web Server Patches

Periodically, the URL http://www.iplanet.com/downloads/download/ should be visited to retrieve the latest service packs and http://www.iPlanet.com/downloads/patches/ should be reviewed for the latest patches. In addition, http://docs.iPlanet.com/docs/manuals/enterprise.html should be reviewed for the latest release notes. Release notes contain important information on various administrative issues. As of this publication date, iWS, Enterprise Edition 4.1 SP5 is the latest version of the iWS, Enterprise Edition server.

Post Installation

Quality Feedback Agent

Future releases of the iWS will include an error-handling mechanism called the Quality Feedback Agent. If the iWS crashes, this agent will automatically send error information (stack and register dump) to the Sun-Netscape Alliance. Although the agent does not send documents, information concerning the state of a web server during a crash may be

deemed sensitive. If the Quality Feedback Agent is invoked on Windows NT, it will send the following information:[1]

- Stack Trace – shows where iWS failed and what functions were called just before failure
- Program Counter
- Registers – provides the state of the processor at the time of the failure
- Dynamic Libraries – shows any additional dlls that might have been running with or missing from the iWS when it failed
- Threads – identifies potential race conditions with other applications or with different processes in the iWS
- OS Version
- Processor Type
- Stack Data – provides top 2048 bytes on the stack; **shows** the value of variables passed into a function that was running at the time of failure

Careful consideration should be made before deciding on whether or not to invoke the Quality Agent.

Administration Server and SSL 3.0

When the Administration Server is accessed remotely, Secure Sockets Layer (SSL) 3.0 should be enabled with the strongest ciphers (Fortezza/128 bit algorithm or higher) available to both the client and Administration Server. SSL requires the administrator to enter a certificate/key pair password through a dialogue box locally on the host, which interacts with the Windows NT desktop. This is required for the Administration Server to start-up. The Administration Server service should be set to "Log On As: System Account" with the "Allow Service to Interact with Desktop" checkbox checked. If the Administration Server is exclusively accessed from the host system then SSL is not necessary.

When SSL is enabled, server responsiveness will be slightly degraded. In addition, by setting this service to "Log On As: System Account," it has access to all system resources just as a Windows NT Administrator. This creates a riskier scenario in the event of an attack. SSL client authentication is supported, i.e., any client certificate that is signed by a trusted certificate authority will allow establishment of an SSL connection; however, the Superuser account is not mapped to the certificate. The Superuser ID and password are still required to access the Administration Server web pages. Lastly, SSL 2.0 should be avoided since there are known vulnerabilities with this version of the protocol, e.g., possible ciphersuite rollback attack.

[1] Taken from the iPlanet Web Server, Enterprise Edition 4.1 Administrator's Guide, March 2000.

iWS and SSL 3.0

The iWS service should be configured based on the desired use of the server. If an iWS is going to be used exclusively for non-secure transactions, the service should be set to "Log On As: This Account: *restricted_user,*" where *restricted_user* is a restricted Windows NT user account as described in the Pre-Installation section.

If iWS is going to be used for sensitive transactions, SSL 3.0 should be enabled with the strongest ciphers available to both clients and the iWS. Unfortunately, the *restricted_user* account that is used to start the iWS service will not work properly with SSL. The release notes for iWS state that work-around (396337) will resolve the problem; however, despite several attempts, the work-around did not resolve the problem. In the case of SSL 3.0, Windows NT requires that the iWS 4.1 service be set to "Log On As: System Account" with the "Allow Service to Interact with Desktop" checkbox checked. This is the same as described above for the Administration Server and requires that the administrator enter the password through a dialogue box on the host system. As stated earlier, if iWS is successfully attacked, the attacker has full system rights. During testing, this configuration functioned without error (i.e. no extraneous errors or web page malfunctions).

iWS Certificate Mapping Configuration File

iWS can allow the mapping of SSL client certificates to their respective Lightweight Directory Access Protocol (LDAP) user accounts.[2] In order to do this, a certificate authority (CA) must be present in the trust database that is trusted for client certificates and an access control list (ACL) must be set up. The ACL basically indicates what method of authentication will be enforced for a given resource—in this case SSL. In addition, a configuration file, similar to Figure 1-1, named certmap.conf located in *server_root/userdb/* is used to determine how a client certificate will be mapped to a user account in the LDAP directory. The basic steps are as follows:

1. Client establishes an SSL connection with the iWS *servername* with a client certificate that is signed by a trusted CA.

2. Depending upon the configuration of the named mapping,[3] iWS uses attributes of the user certificate's subject distinguished name (DN) to map to a branch point in the LDAP directory. In some instances, the certificate may not have attributes suitable for establishing the branch point. In this case, the search is begun at the top of the directory tree. The issuer DN can optionally be used to match to a specific CA.

3. At the branch point, the directory searches for an entry that matches the filter information. The filter information is derived from selected attributes from the subject DN contained in the client certificate, e.g., userid or e-mail address or both.

4. If an LDAP entry is found, optional verification of the client certificate with the certificate entry in the LDAP directory is performed.

---

[2] iPlanet Directory Server was used for testing.

[3] Figure 1-1 illustrates a named mapping. The name of the mapping is testmap. The first line indicates what the issuer DN of a client certificate should contain. The second line indicates where in the directory to search. The third line indicates how to filter. The fourth line indicates that certificates will not be compared to entries theLDAP directory.

The mapping process is fairly straightforward, but depending upon how certmap.conf is configured, it may be possible for an unauthorized user to use a client certificate in a certificate masquerade attack against an iWS user account.

A certificate masquerade can take place in the following situations:

- Arbitrary Client Certificate Attack (ACCA) – An unauthorized client certificate from a trusted CA may have some of the same attribute values that can filter to a unique LDAP directory user account, e.g., the certmap.conf is set up to start at the very top of the LDAP directory tree and only uses the userid of the certificate to filter down to the LDAP entry.

- Masquerading CA Client Certificate Attack (MC3A) - Two or more CAs are trusted by the iWS and one CA creates a subordinate or cross-certified certificate of a peer trusted by the iWS. A masquerading subordinate CA will allow a malicious root CA to match any part of a client certificate except for the subject key identifier, authority key identifiers (assuming identifiers are unique), and the public key as illustrated in Figures 1-2 thru 1-6.

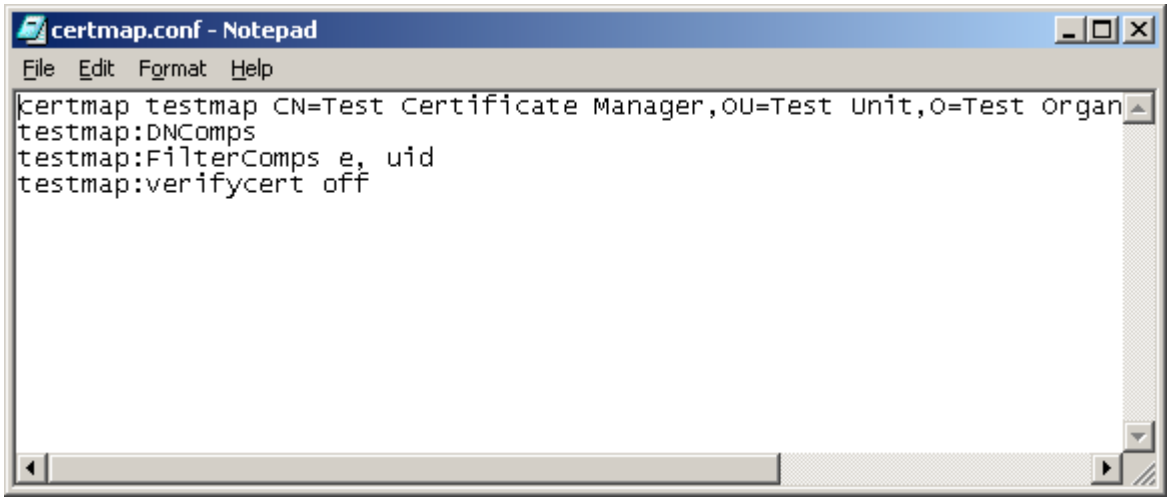

Figure 1-1 A certmap.conf File Entry with Issuer DN[4]

[4] A certmap.conf file determines how a client certificate will be mapped to an LDAP directory user account.

Figure 1-2 Trusted Test Organization Certificate Authority[5]

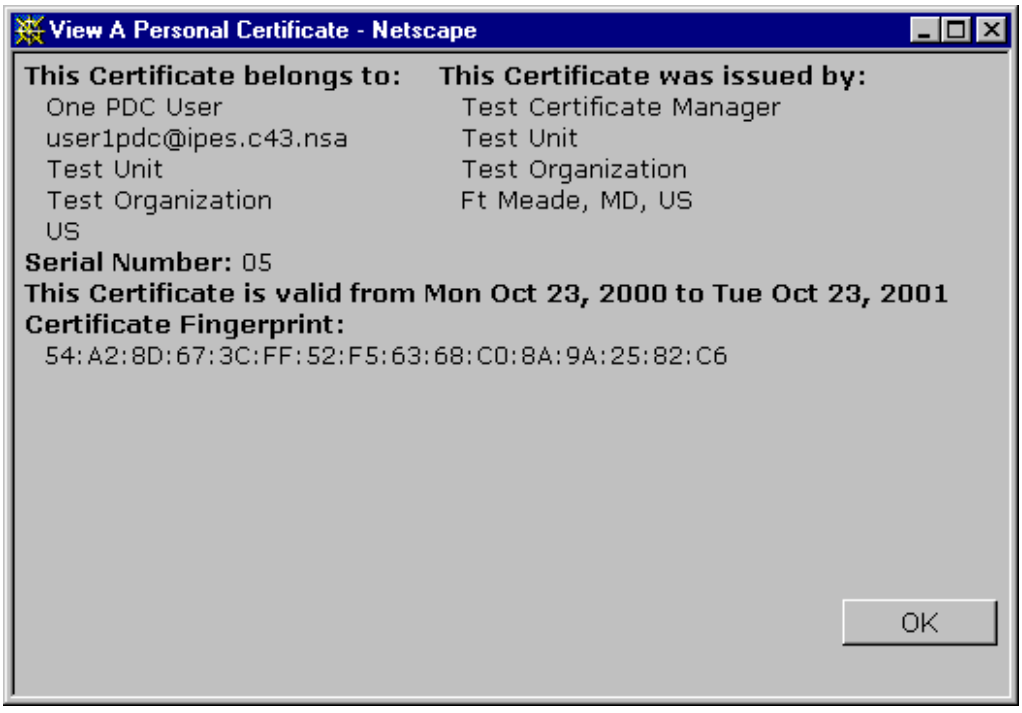

Figure 1-3 Actual Client Certificate for a Valid User[6]

[5] A trusted CA in the test iWS installation.

[6] A client certificate issued by the Test Certificate Manager CA for a user named One PDC User.

Figure 1-4 Trusted Partner Organization Certificate Authority[7]

Figure 1-5 Partner's Masquerading Subordinate Certificate Authority[8]

[7] Partner Certificate Manager is a peer CA of Test Certificate Manager.
[8] A fake Test Certificate Manager subordinate CA created by Partner Certificate Manager CA.

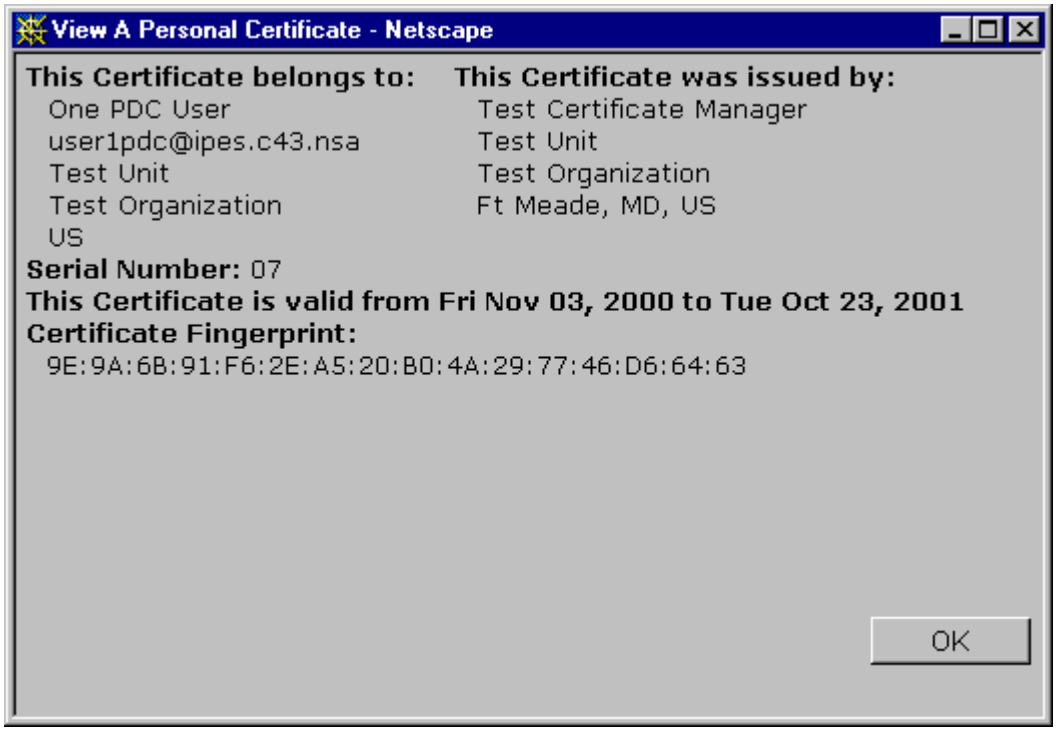

Figure 1-6 Masquerading Client Certificate[9]

In either of the above two cases, turning on the certificate verification option via the certmap.conf file is the simplest method for avoiding a possible client certificate masquerade attack; however, this will require that each user have a corresponding certificate in the LDAP directory so that the SSL client certificate and LDAP directory user entry certificate can be compared. If turning on the certificate verification option is not operationally feasible, then based on the situation, one of the following should be taken into consideration:

1. Assuming that the CA(s) can be trusted not to perform an MC3A and a CA will not inadvertently issue duplicate certificates, then the remaining issue of an ACCA can be addressed by performing the following:[10]

    - A named mapping should be entered in the certmap.conf file that includes the issuer DN. One needs to consider what component(s) of the subject DN will make the mapping unique. If the client certificates have subject DN components, e.g., organization unit, organization, locality, and country attributes, that can establish a branch point in the directory and filtering values, then they should be used. If the CA issues certificates that have unique userids, then they can be used as a filter. It is preferable to include the e-mail address as well.

    - A mapping may need to start at the top of the directory tree, i.e., no branch points possible. This case may arise when clients accessing a given iWS do not have any common organizational attribute values in the subject DN thereby making it

[9] Partner Certificate Manager has created a client certificate (Figure 1-6) that can masquerade as the client certificate created by Test Certificate Manager (Figure 1-3).

[10] Note: CA software will allow the issuance of duplicate certificates, e.g., certificate updates. It is up to the CA agent to prevent any unauthorized duplicated certificates.

impossible to establish a branch point in the directory. It is imperative that the filtering components be unique. Generally, the userid may be unique within an organizational CA but probably not in a commercial CA or across multiple CAs. The use of the e-mail address should make a better filtering attribute.

- If neither of these solutions addresses the situation, then the CmapLdapAttr attribute may need to be established in the LDAP directory. The LDAP directory administrator will need to store subject DNs from all certificates belonging to users in the LDAP directory. The attribute isn't a standard attribute and will require that the LDAP directory schema be extended. Discussion concerning extending the directory schema for iPlanet Directory Server can be found in the *iPlanet Directory Server Administrator's Guide* and the *iPlanet Directory Server Deployment Manual*. The subject DN of the client certificate is used to search the entire LDAP directory for a matching entry, which has a matching attribute value. Before this solution is used, serious consideration should be given to simply collecting all client certificates and storing them in the LDAP directory and using "verifycert on" in the certmap.conf file.

2. If the iWS servername will trust multiple CAs, and if the concern is M3CA, then it is preferable that the CAs issue cross-certified certificates to one another. This will allow the respective CAs to apply a name constraint against each other. In some cases, organizations may not be able to cross-certify, e.g., commercial CAs. In order to restrain them, the iWS will need to have its own custom mapping (see *NSAPI Programmer's Guide for iWS*). In addition to using the subject DN components, the custom mapping would need to compare the fingerprint of the expected issuer against that of the issuer that was used to verify the client certificate. If the match fails, so must the client authentication. The AuthorityKeyIdentifier could be used, but it has two drawbacks. First, the AuthorityKeyIdentifier is not present in X.509 v1 certificates. Lastly, the AuthorityKeyIdentifier may not be unique if it does not include a hash of the CA's public key.

File Permissions

File permissions for all directories and files located on the separate NTFS partition where the iWS and Administration Server reside should be set to allow Windows NT Administrators and members of the iPlanet Administrators Group Full Control. It is recommend that the restricted user account only be allowed Modify permissions as shown in Figure 1-6. In testing, the Modify Permission caused no errors; however, if errors are encountered, then the restricted account should be granted Full Control instead. Other file permissions should remain the same as recommended in the "*Guide to Securing Microsoft Windows NT Networks.*"

iWS uses the NT account that has been set to "Log on As: *restricted_account*" and its associated rights and privileges to manage content. This account acts on behalf of iWS users that have been created by an iWS administrator; however, the Administration Server and iWS enforce the permissions that have been assigned to iWS users and administrators by an administrator. These permissions are set in iWS and Administration Server access control lists (ACLs). Administrators should test their ACLs to ensure they function as intended.

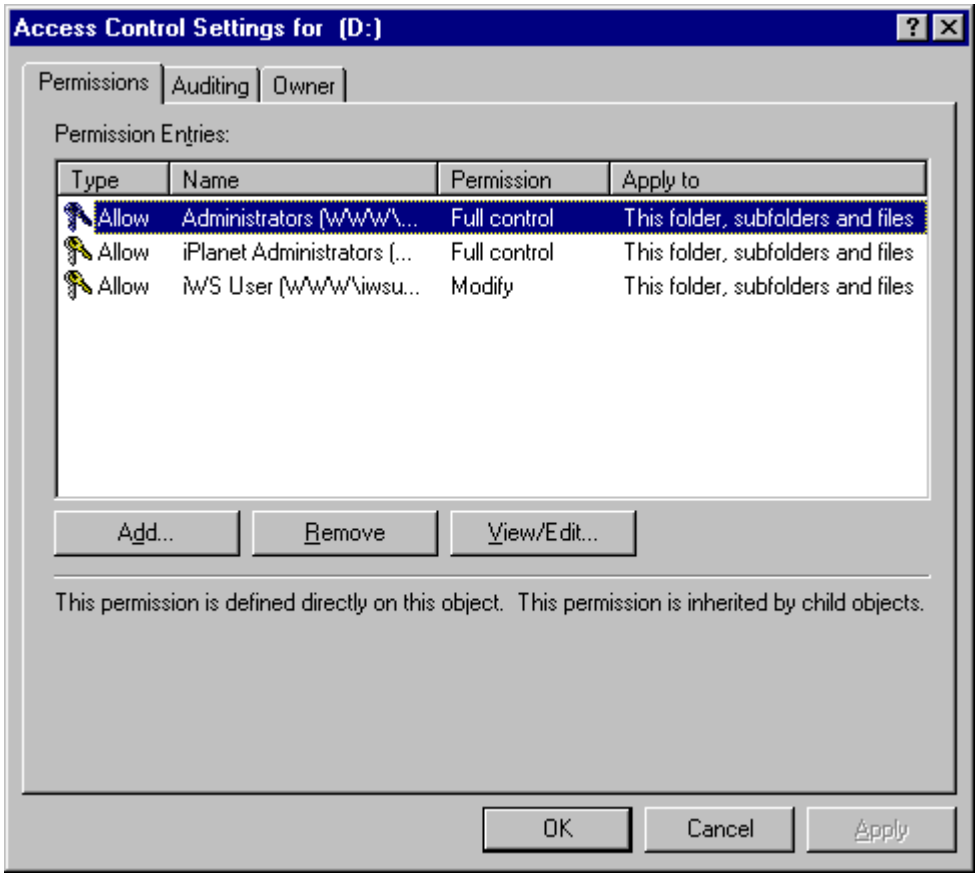

Figure 1-6 File Permission Settings for the iPlanet Partition.

Summary

In summary, when installing the iWS one should:

- Implement recommendations from the "*Guide to Securing Microsoft Windows NT Networks.*"
- Create an iWS restricted user account with the Windows NT rights "Log on as a Service" and "Log on Locally."
- Create a Windows NT iPlanet Administrators Group.
- Install iWS on a separate partition of a non-PDC/BDC server.
- Establish an Administration Server Superuser account with a robust password and at least somewhat unpredictable account name. Password should not be the same as NT administrator.
- Assign an unused random port above 1024, but excluding port 8888.
- Apply all Administration Server and iWS patches.

- Run the Administration Server using the System Account. If it is necessary to access Administration Server remotely, enable SSL 3.0. Use strongest ciphers possible (Fortezza or 128 bit algorithm or greater). If possible, use client certificates for authentication.

- If an iWS is used for sensitive transactions, run the iWS service under the System Account and turn on SSL 3.0. Otherwise, run the iWS service under the *restricted_user* account without SSL 3.0.

- If possible, when using client certificates for authentication in iWS, load client certificates to the directory and set the verifycert directive to 'on' in the certmap.conf file.

- Set iWS and Administration Server file permissions as specified.

Chapter
2

iPlanet Web Server Administration Server

Introduction

The Administration Server manages all iWS instances. It is automatically included in the iWS installation and is one way to administer the iWS. Netscape Console can also be used, but will not be covered. The Administration Server requires an appropriate configuration to minimize security vulnerabilities, as does the iWS, which will be detailed in the next chapter. This chapter will describe Administration Server settings that require special attention due to their security relevance.

Administration Preferences

Superuser Access Control

The Superuser Access Control (SAC) option, available through the Administration Server Preferences tab, allows the Superuser account access locations to be set. The username and password were originally set during iWS installation. The Superuser username and password are required for any remote or local administration tasks as well as for creating subgroups of administrators.

Due to its authority over all the iPlanet servers, the Superuser login is especially important and its use should be further restricted through access control measures. Permitted hostnames and IP addresses from which the Superuser account can be accessed should be specified as shown in Figure 2-1. If remote administration is not used, then the SAC should be configured with the local hostname and IP address.

The default installation of the Administration Server will allow access from any host on the network. If remote administration needs to be invoked, then the IP(s) and the hostname(s) should be set for the remote workstation(s) that will be used for administration. In addition, SSL 3.0 should be activated with strong ciphers. Great care should be taken when using wild cards for locations, e.g., 164.120.65.*, as this could allow several workstations to access the Administration Server. Also, the physical security of these workstations should be reviewed so that physical access is limited to actual administrators.

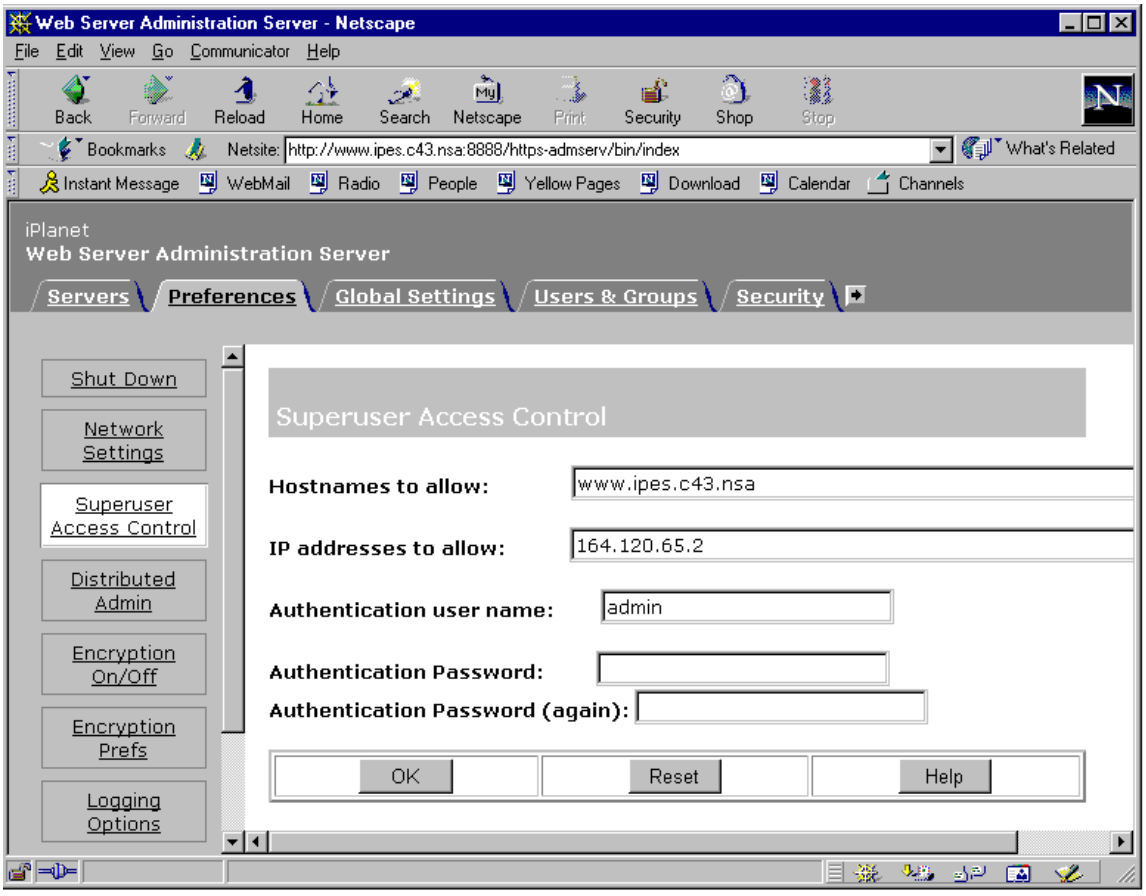

Figure 2-1 Superuser Access Control Settings.

Table 2-1 is a guideline for controlling the use of the Superuser account for accessing the Administration Server.

	Local Administration Only	Administration from inside/outside an Intranet
Block request on router/firewall for SAC administration port.	X	
Specify the local host name and IP in the SAC.	X	X
Change the default HTTP port from 8888 to some other unused port. In addition, the port should be changed periodically. Also, the default Superuser account name should be changed.	X	X
Create ACL on router/firewall indicating authorized external hosts for the administration HTTP port. [11]		X
List specific remote hosts in SAC page. This list should match the hosts in the ACL of the router/firewall.		X

Table 2-1 Router/Firewall Superuser Access Control List Settings.

Distributed Administration

Distributed administration allows other administrators to control and configure the Administration Server and iWS web server instances; however, it is strongly recommended that this option not be activated. If activated, administrators should have both a user account and distributed administrator account. The accounts must be created in the LDAP directory and can be created by the Administration Server administrator or the directory administrator. Any additional administrators must be placed in the iPlanet Administrators group.

Encryption On/Off

Netscape's strongest security feature is the capability to use encrypted communications between a remote client and the Administration Server. Public key encryption is implemented through the SSL protocol, which resides between the application layer and TCP/IP providing end-to-end security. The additional security offered by SSL comes at a cost to performance. Due to encryption and decryption processing, transfer rates may be reduced slightly.

[11] If Distributed Administration is turned on, then the ACL should include the hosts for authorized distributed administrators.

SSL is enabled through the Administration Server's Encryption On/Off option of Preferences. An alias is required for successful SSL configuration. An alias is the name associated with an installed server certificate to be used for a given secure port (see the section entitled "Security" which follows). Once SSL has been enabled on the Administration Server, it is activated when a client URL requests access to the administration web page on the server and port using https://. It is strongly recommended that SSL 3.0 be enabled on the Administration Server if it is to be accessed remotely. Otherwise, SSL is not necessary.

Encryption Preferences

During the initial stages of establishing an SSL session, the cipher with the largest encryption key is negotiated for use in that session. The cipher must be common to the Administration Server and the client's browser. The Encryption Preferences option of the Administration Server's Preferences allows the administrator to select those ciphers that are to be supported during an SSL 2.0 or SSL 3.0 session. Due to security vulnerabilities in SSL 2.0, it is recommended that only "SSL 3.0" be selected under the "Allow" option. In addition, only the ciphers listed in Figure 2-2 should be activated. This will allow the Administration Server to negotiate the largest encryption key size possible by the client browser for accessing the administration web pages.

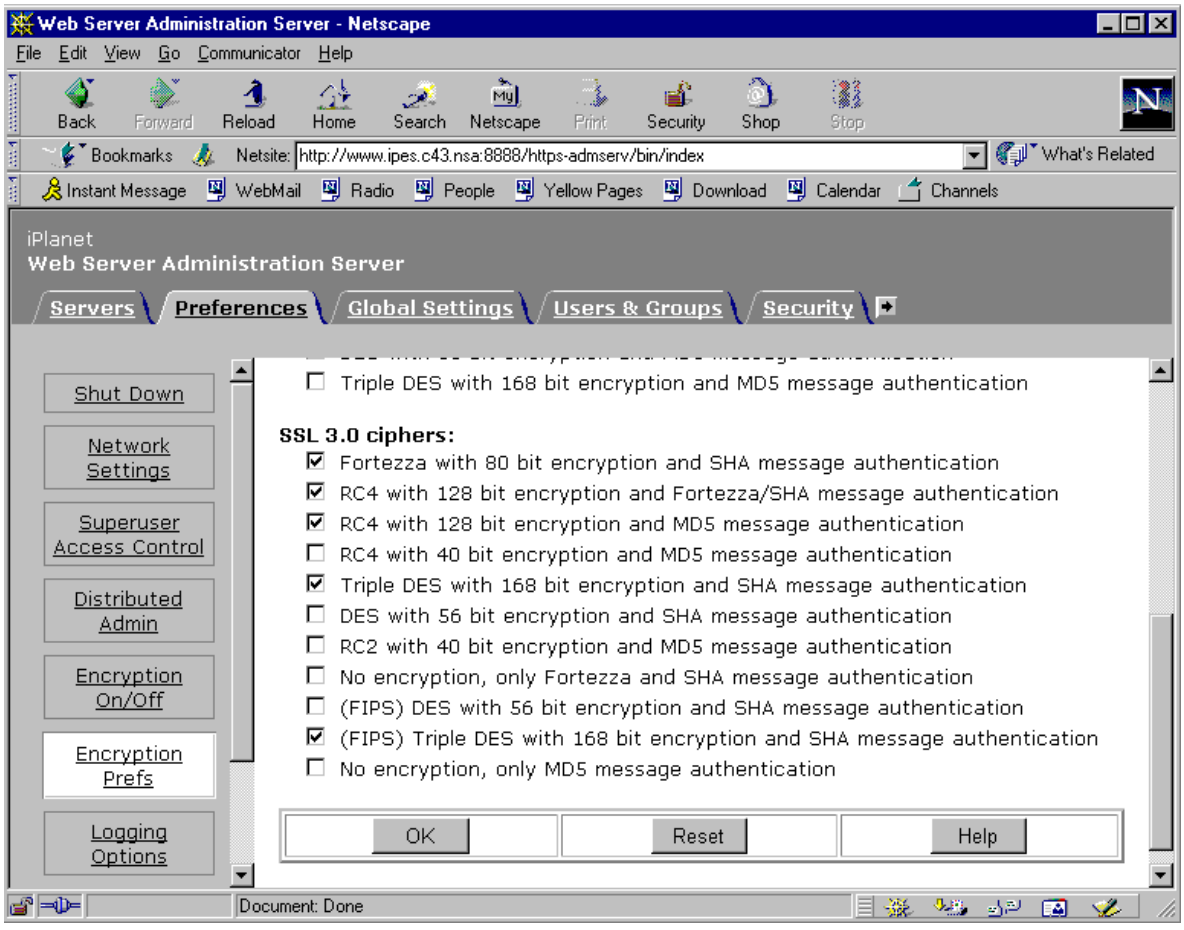

Figure 2-2 Encryption Preferences

Logging

Within the Preferences tab of Administration Server, Logging Options and the capability to view access and error activity are available on the Administration Server. Logging all activity is a critical part of securely administering the server. The log files may provide helpful insight into determining if an attack is active or has already occurred and should be reviewed in conjunction with the operating system logs. Likewise, logging is available on iWS and will be discussed in the Status section of Chapter 3.

Under the Preferences tab, Logging Options allows the administrator to specify the location of the access logs. The default access and error log location is [Application Root]\Netscape\Server4\https-admserv\logs. The error and access logs can be reviewed through the View Error Log and View Access Log options available in Preferences or a text editor. iPlanet does not support logging to database file formats. The information logged about the client, such as the originating domain name or IP address, time of access, content accessed, status and user authentication information, will greatly help an administrator detect malicious activity and protect against future attacks. It is strongly recommended that the log files specified under Logging Options be monitored on a regular basis and that Windows NT log files be monitored as well. Table 2-2 shows a list of possible items to look for:

Access Log Entry	Error Log Entry	Possible Security Relevance
HTTP Status 401	Security – "user *Superuser* password did not match pwfile" – this entry will correspond to a related HTTP 401 Access Log entry	May indicate attempts of unauthorized user access; several attempts could indicate an attempted breach in security. Note such an attack could be periodic or intermittent, thus it is important to maintain archives of logs.
HTTP Status 403		Could be an indicator of possible attempts to unauthorized directories or files.
HTTP Status 404		Indicates the URL requested was not found. May indicate possible buffer overflow attack if the URL is over a 1000 characters.
Paired HTTP 401 and HTTP 500		Entries may indicate attempts to logon to the Administration Server from an unauthorized client with a valid account and password.

Table 2-2 Administration Server Logs – Possible Suspicious Entries.

Global Settings

The Restrict Access option available through the Global Settings tab of the Administration Server is intended to establish ACLs for controlling administrative access to the Administration Server and iWS. In order to make any settings, Distributed Administration must be turned on. As stated earlier, it is strongly recommended that Distributed Administration not be turned on.

Users & Groups

The Users & Groups tab provides the administrator with the capability to specify users who are permitted to access Administration Server and iWS resources. User information such as full name, user ID, password and email address can be entered. In addition, users can be added to Groups based on common access requirements for web content or services.

This option is important for specifying detailed ACLs established through the Restrict Access setting of Server Preferences tab for iWS and Global Settings tab for Administration Server. Within an ACL definition, fields for users and groups can be used for granularity and thus provide a greater level of security.

Security

The Security tab of the Administration server is where all certificate and key management functions are performed. In order for SSL to function correctly, a signed server certificate must be installed and associated with an alias name. Functions such as creating a trust database for certificates, requesting/installing certificates, and password changes are managed from this tab and are shown in Figure 2-3.

Within this category, the Change Password option is of security importance. This password is required to enable SSL on the specified server. It must be carefully protected from intercept and should not be changed over a non-secure network connection. SSL should be enabled on the Administration Server before making any changes over a remote connection.

If client authentication is enabled, only trust those CAs that will be needed to validate client certificates. The Administration Server's default settings are for all CAs to be marked as untrusted. Likewise, any client browsers that will be used for remote administration should only have the respective CA certificates trusted for validation of the Administration Server certificate and any other web server certificates that are needed for administration purposes. Care should be taken when installing CA certificates. The certificate fingerprint for a CA should be obtained through a secure out-of-band method and used to validate a CA certificate before adding it to the trust database.

Precautions should also be taken when acquiring certificate revocation lists (CRLs). CRLs should be acquired through a secure channel, e.g., SSL. A secure method is especially needed where CRLs are expected to have a long period of time between updates. This suggestion is made so that replay attacks may not be initiated, i.e., new CRL is issued between updates and an attacker replays a previous list that will be

consider valid by the iWS. In cases where the CRL lifespan is short, a secure channel may not be as critical. As of iWS 4.1 SP4, the Manage CRL's option was not functioning. In any case, it is recommended that revocation be handled via an LDAP directory that stores certificates and where a CA frequently updates the directory with certificate entries. In addition, certmap.conf mappings will need to contain a directive that specifies that the certificate verify option is on. See iWS Certificate Mapping Configuration File section.

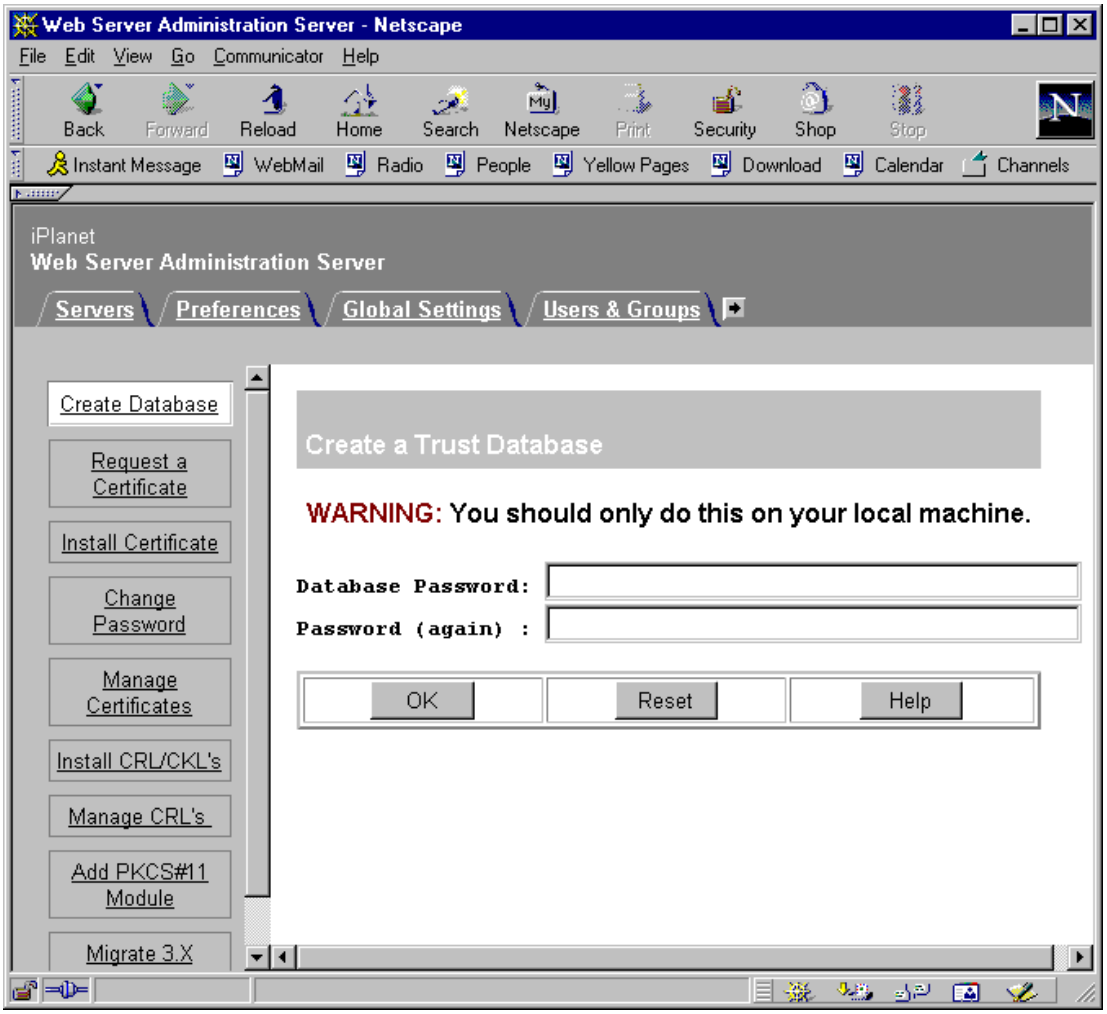

Figure 2-3 Administration Server Security Tab.

Summary

The Administration Server provides several security options to restrict access to the administration web pages for the Administration Server and iWS. The following points are recommended for implementation on the Administration Server:

- Restrict the Superuser account by specifying hostnames and IP addresses from which a connection is allowed. If local administration, specify the host's hostname and IP. Ensure proper router/firewall configuration.

- Do not use Distributed Administration.

- Enable SSL 3.0 with the strongest cipher possible when remotely accessing the Administration Server. Select only SSL 3.0.

- When possible, set the Encryption Preferences tab as illustrated in Figure 2-2.

- Review the access and error logs frequently to ensure that no suspicious activity is occurring on the Administration Server.

- Assign Users to Groups based on requirements. For example, Users in the Human Resources Group may only need to access personnel resources while the Engineering Group may only need to access research and development content.

- Trust only those CAs that are necessary for client authentication.

- Use a secure channel to obtain CRLs, especially in environments where there is a long period between updates.

Chapter

3

iPlanet Web Server, Enterprise Edition

Introduction

iWS provides a complete line-up of services to satisfy both intranet and Internet requirements. iWS offers static HTML web pages to dynamic content customized through programming languages such as Java, C++, and JavaScript.

This chapter will describe the iWS services that require special security consideration. The following recommendations are provided to make these services more secure.

Server Preferences

Dynamic Configuration Files

Dynamic Configuration Files are an access control mechanism which create restrictions on resources based on settings defined in a configuration file. iWS supports *.htaccess and *.nsconfig files; however, there is no support for LDAP. Basically, there is a set of "directives" which iPlanet permits an administrator to use for purposes of defining nearly all the same parameters as available from the Restrict Access (next section) web page. The configuration file can be quite complex. It is recommended that resources be restricted according to the approach outlined in the Restrict Access section.

Restrict Access

The Restrict Access option of iWS Server Preferences is one of the most important security features of the server. From this option, all ACLs are managed for the server resources. ACLs may be established or edited based on a variety of control parameters. iWS resources can be restricted using control parameters such as allowing or denying specific users or groups based on an originating IP address or hostname. In addition, more detailed restrictions based on file permissions and customized instructions may also be implemented.

The restrict access option may be used on directories or individual files. There are three options available from the Restrict Access web page. The first option, "Pick a resource," allows a particular server directory or file to be selected for applying the defined ACL. The second option, "Pick an existing ACL," provides quick access to an existing ACL through a pick-list. Once the ACL is selected, the restrictions may be modified. Finally, the third option, "Type in the ACL name," allows the administrator to give the ACL a user-friendly name. This option requires modification of the `obj.conf` file and a thorough understanding of how .acl files function to work correctly. Just a word of warning:

Defining ACLs can be tricky and can actually produce unexpected results if one doesn't read the Help section carefully. It is recommended that only the first two options be used. In addition, ACLs should be tested. Figure 3-1 shows an ACL implemented on an iWS resource. The ACL displayed restricts iWS access to the group iPlanet Users.

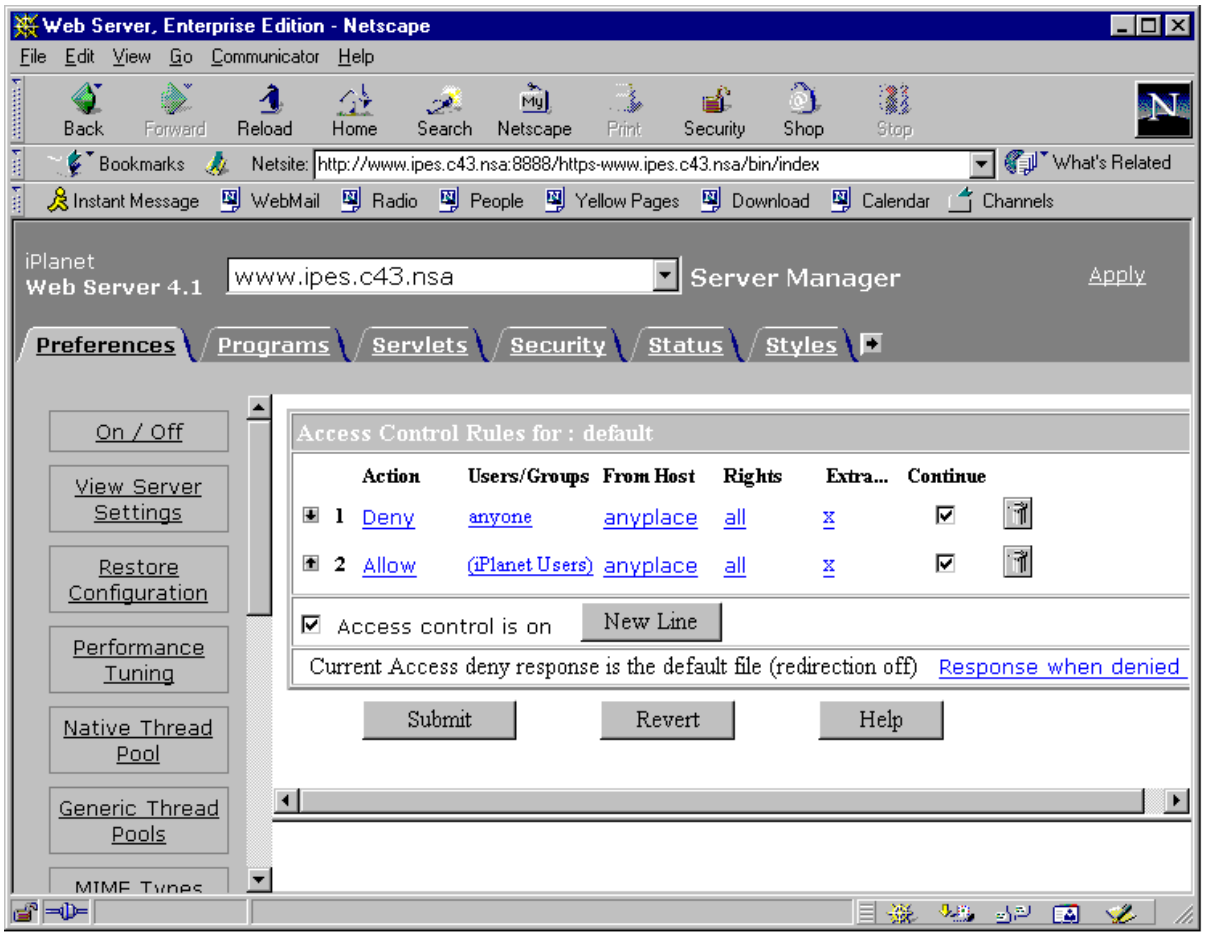

Figure 3-1 iPlanet Web Server's Restrict Access Option.

Encryption On/Off

iPlanet's strongest security feature is the capability to use encrypted communications between iWS and the client. Public key encryption is implemented through the SSL protocol, which resides between the application layer and TCP/IP providing end-to-end link security. The additional security offered by SSL comes at a cost to performance. Due to encryption and decryption processing, transfer rates may be reduced slightly.

SSL is enabled through the Encryption On/Off option of Server Preferences. A secure port number and alias are required for successful SSL configuration. An alias is the name associated with an installed server certificate to be used for a given secure port. Once SSL has been enabled on iWS, it is activated when a client URL requests a resource to be served from that server and port using https://. It is strongly recommended that SSL be enabled on iWS for communications involving sensitive information.

Encryption Preferences

During the initial stages of establishing an SSL session, the cipher with the largest encryption key is negotiated for use in that session. Generally, the larger the encryption key size used by the cipher the more difficult the encryption process is to break. The cipher must be common to iWS and the client's browser. The Encryption Preferences option under Server Preferences allows the administrator to select those ciphers that are to be supported during a session. The administrator may select the level of security implemented by the cipher and its key size to be supported by iWS. In addition, the Stronger Ciphers option under Server Preferences allows specific iWS resources to be accessible only by a client's browser supporting a selected cipher key size of 40 or 128 bits. In other words, it restricts access by clients that do not use a key length equal to or exceeding this key length. Key lengths of 128 bits or more and SSL 3.0 should be used.

Programs

This document focuses on securely configuring the web server but does not delve into many aspects of the development of secure web applications. Much open source material is available on this subject. One reference we have found as particularly useful is *Designing Secure Web-Based Applications for Microsoft Windows 2000* by Michael Howard.

Common Gateway Interface

Common Gateway Interface (CGI) programs provide additional functionality to web content. They are written in various languages such as C, C++ programs and Perl scripts. A CGI program accepts input parameters and generates results that are returned to the web server. Typically, CGI programs are executed in response to a command originating from an HTML form but may also be executed directly through its calling URL.

iWS supports CGI programs through two different configurations. The first configuration, as shown in Figure 3-2, specifies a CGI directory that will be used as a central repository for all CGI programs. The iWS administrator is responsible for installing programs and setting appropriate user permissions on the programs. The main security advantage to this configuration is its centralized administration. Administrators can reduce the risk of CGI related risks by requiring developers submit their programs to them for review and installation. All files in the specified CGI directory are expected to be CGI executables.

iWS supports another CGI configuration based on CGI file types. This permits developers to create a CGI program with a particular file extension that is recognized by iWS. The program may reside anywhere in the document root on the server. It provides greater flexibility for developers but creates more security challenges for the iWS administrator. This configuration allows CGI programs to exist in several possible locations along with other HTML files on the server and, therefore, is more difficult to securely administer. This configuration is not recommended. Figure 3-3 shows the CGI file type configuration.

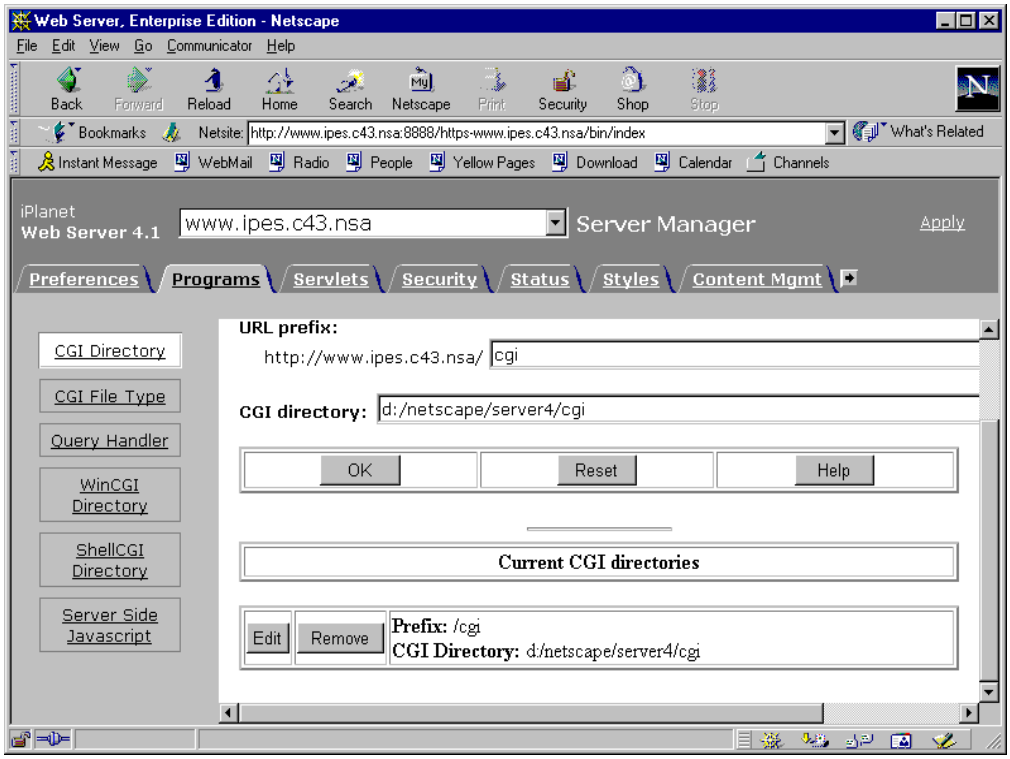

Figure 3-2 iPlanet Web Server CGI Directory Configuration (Recommended).

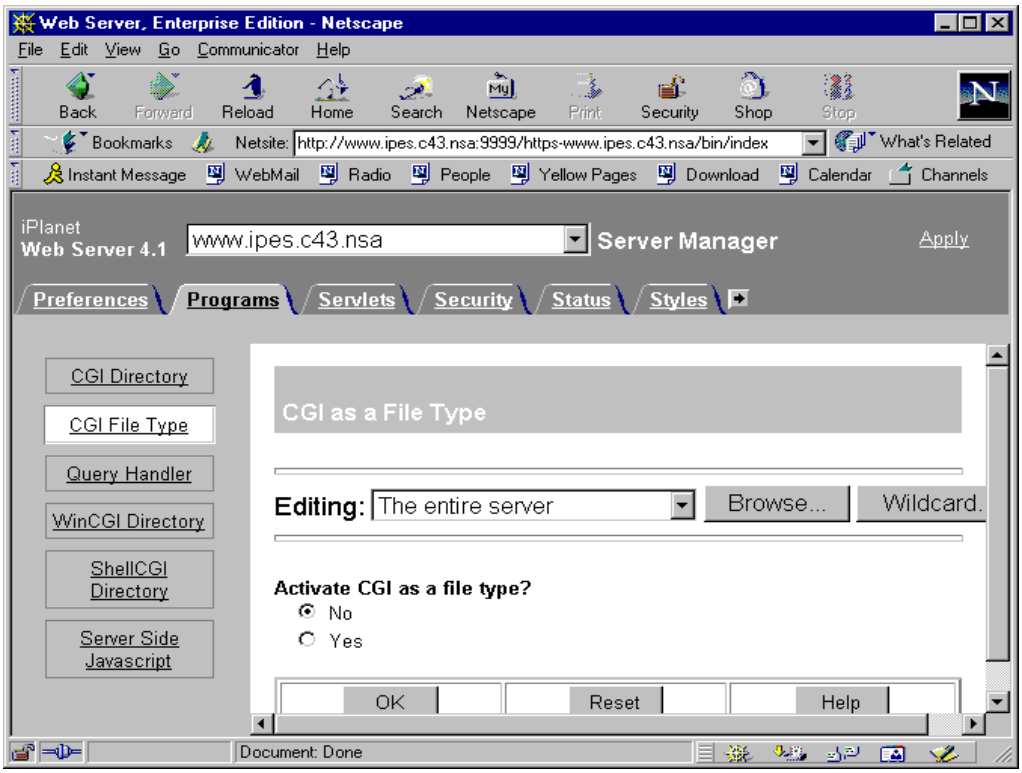

Figure 3-3 iPlanet Web Server CGI File Type Configuration (Not Recommended).

Regardless of the type of configuration used, CGI programs create several security vulnerabilities. Some general security guidelines for using CGI should be followed. Never put a script interpreter in the CGI directory. One of the more common security breaches occurs when web server administrators place CGI script interpreters such as Perl.exe in the CGI directory or another directory readable via HTTP. It is normally done for ease of use but creates an enormous hole in the security of the system for obvious reasons. Commands can be executed by appending them to a URL that first calls the Perl.exe interpreter in the CGI directory.

In addition, specific Windows NT file permissions should be set on the CGI directory containing the programs. For example, only administrators should be permitted to write to a centrally located CGI directory. If the CGI file type configuration is used, only owners of the specific directories should be permitted to write to the directory. Further access restrictions available through the Restrict Access option under Server Preferences may be used on CGI directory and files. However, most of the security vulnerabilities associated with CGI programming comes from bad programming techniques. It is extremely important that thorough program testing and evaluation be performed to avoid the introduction of security vulnerabilities on the server. Some issues to keep in mind:

- If possible, run the web server using the restricted user account.
- Use a CGI directory versus CGI file type.
- Place interpreters in separate directories.
- Programs should:
    - edit form inputs before usage, e.g., type checking to determine if the value entered is proper for the variable that it is supposed to be assigned to;
    - edit form inputs for attempts of buffer overflows, e.g., user attempts to assign a string of 100 characters to a variable that is defined to hold 20;
    - edit variables to determine if they contain commands, e.g., if an interpreter is being used, does an input character or combination of characters create commands that the interpreter may interpret as a valid command or instruction.

WinCGI

Windows CGI (WinCGI) is an interface specification for Windows based web servers. It is similar to CGI with a few exceptions. WinCGI program names end in “exe” and use a different approach to accepting input parameters. Generally, the programs are written in Visual Basic, C, or C++. As with CGI scripting, WinCGI programs are called by the HTML POST or GET methods or through a URL reference (HREF) in an HTML document. Once a WinCGI program is called, information is sent to the program differently than a normal CGI program. A WinCGI program uses a temporary file for passing in parameters needed by the program. The server automatically destroys the temporary file unless script tracing is enabled under the server programs options for WinCGI Directory. Script tracing will save the parameter file in /temp. See Figure 3-4 for WinCGI configuration settings.

Like CGI, WinCGI is capable of executing programs on the web server from HTML forms and, therefore, must be properly configured for safe use. It is recommended that all WinCGI programs be kept in a common WinCGI directory on the server. The directory should then be restricted to allow write access by only specifically permitted users and administrators through Windows NT file permissions. In addition, the Restrict Access option under Server Preferences on iWS may be used to further restrict access to the WinCGI directory and files.

Generally, WinCGI programs are considered safer than scripts because the source code is not available on the server. However, as with scripts, vulnerabilities may exist in compiled WinCGI programs that are not carefully developed. With scripts being interpreted, the source code resides on the server and could possibly be retrieved. Several holes have been detected in both CGI scripts and WinCGI programs due to careless development techniques. See the Common Gateway Interface section for suggestions.

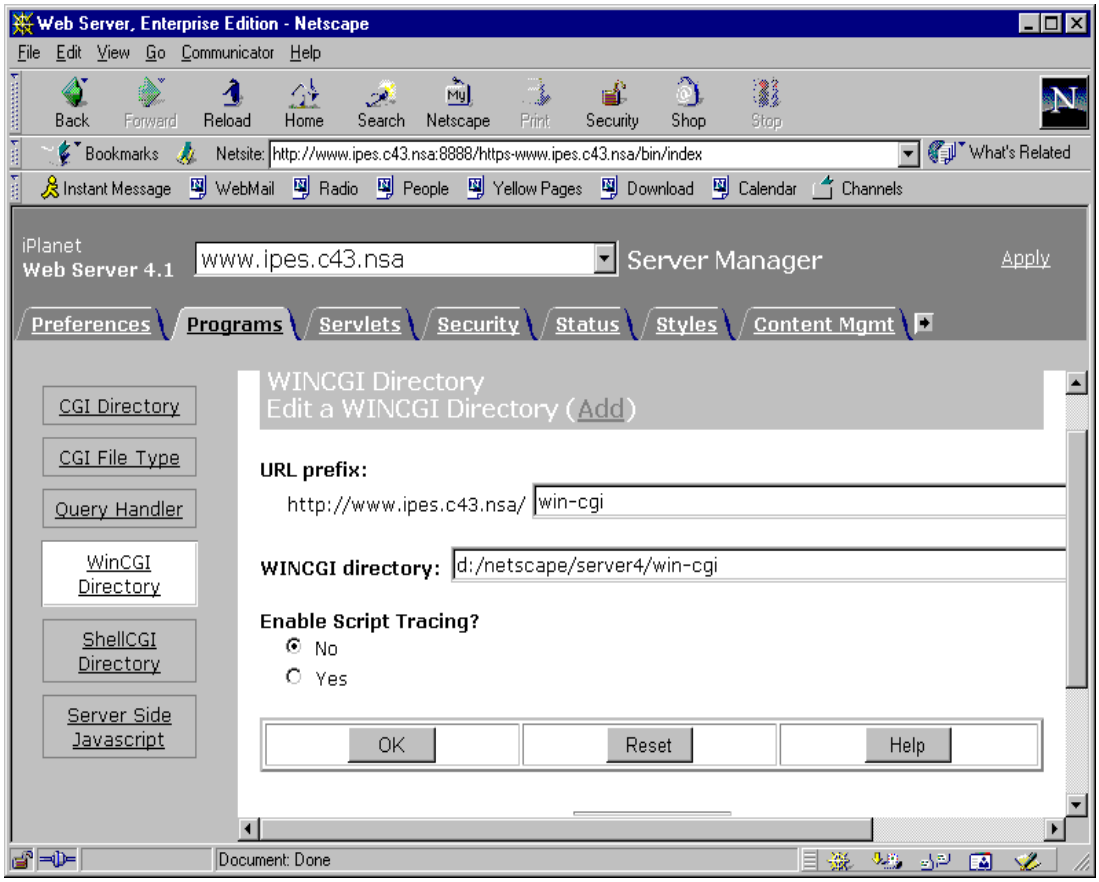

Figure 3-4 iPlanet Web Server WinCGI Settings.

Web Applications Interface

The Web Application Interface (WAI) extends the functionality of iWS through the use of server plug-ins and applications. WAI is based on the Common Object Request Broker Architecture (CORBA), which outlines how software objects communicate over networks of different platform types, operating systems and programming languages. WAI applications and plug-ins for iWS can be developed in Java, C, or C++ to serve HTTP

requests from a client browser. WAI applications run as independent processes on the server while plug-ins run as part of the iWS process. Both require initialization and registration with the server's name service in order that an association may be made between a client's URL request and the compiled program residing on the server. Because these applications are executables residing on the server and can be executed remotely, naturally there are security issues. iPlanet has described several security concerns and has recommended guidelines for safer use of WAI plug-ins and applications.

First, it is possible for a WAI application to become registered with the same service name as an earlier one and run in its place. If the original registered program crashes or for some reason stops running, then another program with the same service name and different functionality may be run instead. The server will only look for a service name match to respond to a request. Second, a user with the appropriate permissions to the folder where plug-ins or applications are kept on the server could write a new file over an existing one. Thus, the same registered name would apply but the application would be different. Finally, a user could register a WAI application that exists on a remote system. By default, iWS listens for Internet Inter-Object Request Broker (ORB) Protocol (IIOP) requests from the local host (127.0.0.1). However, the request may be originating from another machine if configured as described in the third bullet below thus causing iWS to look to other systems for the WAI application. (IIOP is also defined by CORBA to describe how objects interact across networks. IIOP runs on top of TCP/IP). The security concern is that if the external machine permits anyone to write to it, then iWS will be registering external WAI applications possibly without any security controls in place.

iPlanet recommends the following to make WAI application usage safer:

- Restrict remote user access to the iWS host machine. This will prevent an unauthorized program from being written to the host machine and, subsequently, registered and run under a previous service name. Details on restricting remote access can be found in the Remote Access Service (RAS) section of the "*Guide to Securing Microsoft Windows NT Networks*."

- Restrict write access to iWS folders where WAI applications reside. This will protect against new applications being added, registered under a new service name and run. Allow Administrator full control of the WAI folder on Windows NT. This will protect existing WAI applications from being overwritten. In addition, allow the iWS User Account Execute permission only. iWS ACLs are not applicable to scripts and, therefore, do not add any extra security in this particular case.

- Restrict WAI applications to run only on the iWS host machine. iWS assumes that WAI applications will be run on the local host where it resides. However, if IIOP has been configured, then WAI applications located external to the local host will be permitted to register and run through iWS. The external machine hosting the WAI application may not be securely administered, thus providing an opportunity for hostile code to exist. In order to avoid this vulnerability, ensure that iWS is configured to listen for requests only from the local host (127.0.0.1). Specifically, review the `obj.conf` file located in the config folder of iWS and verify that no Init directives calling the IIOPinit function exist.

WAI is supported in iPlanet 4.x versions, but is not guaranteed to be supported in future versions. iPlanet recommends that developers do not develop new WAI applications.

For additional information on using WAI, visit the iPlanet documentation web site [IP4] and see "*Writing Web Applications with WAI*."

Java Servlets and JavaServer Pages

A popular alternative to CGI programming on iWS is Java Servlets (JS). It implements Sun's Java Servlet API 2.2.1. Servlets are executed through a browser's URL call and run on the server's Java Virtual Machine (JVM). Servlets can be used for many tasks including traditional CGI programming, supporting database connectivity, loading a file from the client, and administrative purposes. In addition, servlets offer many advantages over CGI programs including faster throughput and response, better memory management, "write once, run anywhere," and the security features associated when running within the restrictions of the Java Security Manager.

As with all the server-side programming options supported by iWS, vulnerabilities may exist if poor Java programming techniques are used; however, Java's design includes built-in security features, which other programming languages do not have. These features have to be implemented by the programmer. By using the SecurityManager class, digital certificates and access controllers, security vulnerabilities on iWS may be greatly reduced. For additional information on access controllers and the other topics discussed in this section, please refer to "*Java Servlet Programming*" by Jason Hunter [Hun].

A JavaServer Page (JSP) is not a complete Java program like Servlets. JSP is an extension of the Java Servlet technology. A JSP is more like an HTML page, but includes JSP tags that allow the implementation of dynamic content. JSP can be used to provide property values or simple conditionals in a JSP page. iPlanet supports JSP API 1.0. For additional information on using JSP, vist the Sun web site and see *"JavaServer Pages ™ Technology"*.

Server-Side JavaScript

iPlanet's server-side JavaScript (SSJ) is an object-oriented scripting language that provides extensions to iWS. Security vulnerabilities exist in JavaScript as with the other programming capabilities supported by iWS. JavaScript statements may perform relational database functions, support JavaScript and Java interactions and provide access to the server file system. The Application Manager is used to manage SSJ.

The Application Manager is used to install, start, stop, restart, run, debug, and modify SSJ applications. One of the major concerns is using the Application Manager remotely. While access to the Application Manager can be authenticated (Program/Server Side JavaScript) it uses the same login and password as the Administration Server. The login and password are passed in the clear over the network. Therefore, it is recommended that SSL be enabled before using the Application Manager remotely. As was the case with CGI and WAI executables, it is important to restrict write access to directories containing Java applications to help ensure that malicious code is not placed on the server.

Status

Within the Status tab of iWS, an extensive list of options available for logging activity on the server is provided. Logging all activity is a critical part of securely administering iWS. If an attack occurs, the log files may provide helpful insight into understanding the details of the attack. Under the Status tab, Log Preferences provides various settings to help the administrator to actively manage accesses to the server.

There are multiple formats for the log file including the default, individually selectable and customizable. Some of the information provided about the client such as the originating domain name or IP address, time of access, content accessed, status and user authentication information will greatly help an administrator detect malicious activity and protect against future attacks. It is strongly recommended that the Common Logfile Format be enabled under Log Preferences. Also, the default log file name and location should be changed to reside in another local directory. Full Control access should be limited to Administrators, iPlanet Administrators, and the restricted user account.

In addition, the log analyzer can be used to generate statistics about the server, such as a summary of activity, most commonly accessed URLs, times during the day when the server is accessed most frequently, and so on. When possible, the security officer should monitor this information from the Generate Report form to detect any intrusion attempts into the server. The “Archive Log Files” should be setup to save old log files for future reference in the event of an unauthorized attempt is discovered at a later date. Server usage can be monitored during real time with the “Monitor Current Activity” option. Table 3-1 shows the HTTP 4XX and 5XX codes that should be scrutinized in the Access Log file.

Access Log Entry	Possible Security Relevance
HTTP Status 401	May indicate attempts of unauthorized user access; several attempts could indicate an attempted breach in security. Note such an attack could be periodic or intermittent, thus it is important to maintain archives of logs.
HTTP Status 403	Could be an indicator of possible attempts to unauthorized directories or files.
HTTP Status 404	Indicates the URL requested was not found. May indicate possible buffer overflow attack if the URL is over a 1000 characters long.

Table 3-1 iWS Access Log – Possible Suspicious Entries.

Content Management

Additional Document Directories

In the Content Management tab, iWS provides the flexibility to define new document directory mappings. Document directory mappings are used to assign a URL prefix to a physical directory path. The URL prefix then provides access to the files in the physical directory on the server. By default, iWS provides several predefined mappings to support online help, servlets, web publisher and other iWS features. Access to the directories created by default mappings and a newly created mapping is initially unrestricted. Care must be taken not to specify any directories that contain system files or other information that should not be accessed. The administrator must also restrict access to these directories and files by using ACLs. Sub-directories under this directory are subject to the same precautions. Otherwise, files may be freely modified.

Remote File Manipulation

The Remote File Manipulation option under the Content Management tab permits users to perform file management tasks on files residing on the server. Figure 3-5 shows the Remote File Manipulation option in iWS. Enabling file manipulation should be done in conjunction with iWS ACLs. When Remote File Manipulation is enabled, Web Publisher (discussed in the next section) must be disabled. Neither will function correctly if the other is also enabled. ACLs should be established within iWS to ensure that users can only manipulate the files which policy allows. Without iWS or Windows NT ACLs, a user can easily open another user's file, modify it, and re-publish it. It is critical to immediately establish user ACLs when enabling Remote File Manipulation. It is also recommended that content be pushed to the web server via a "drop-box" directory rather than directly to a given directory. This will allow the administrator to review web content before making it available to users.

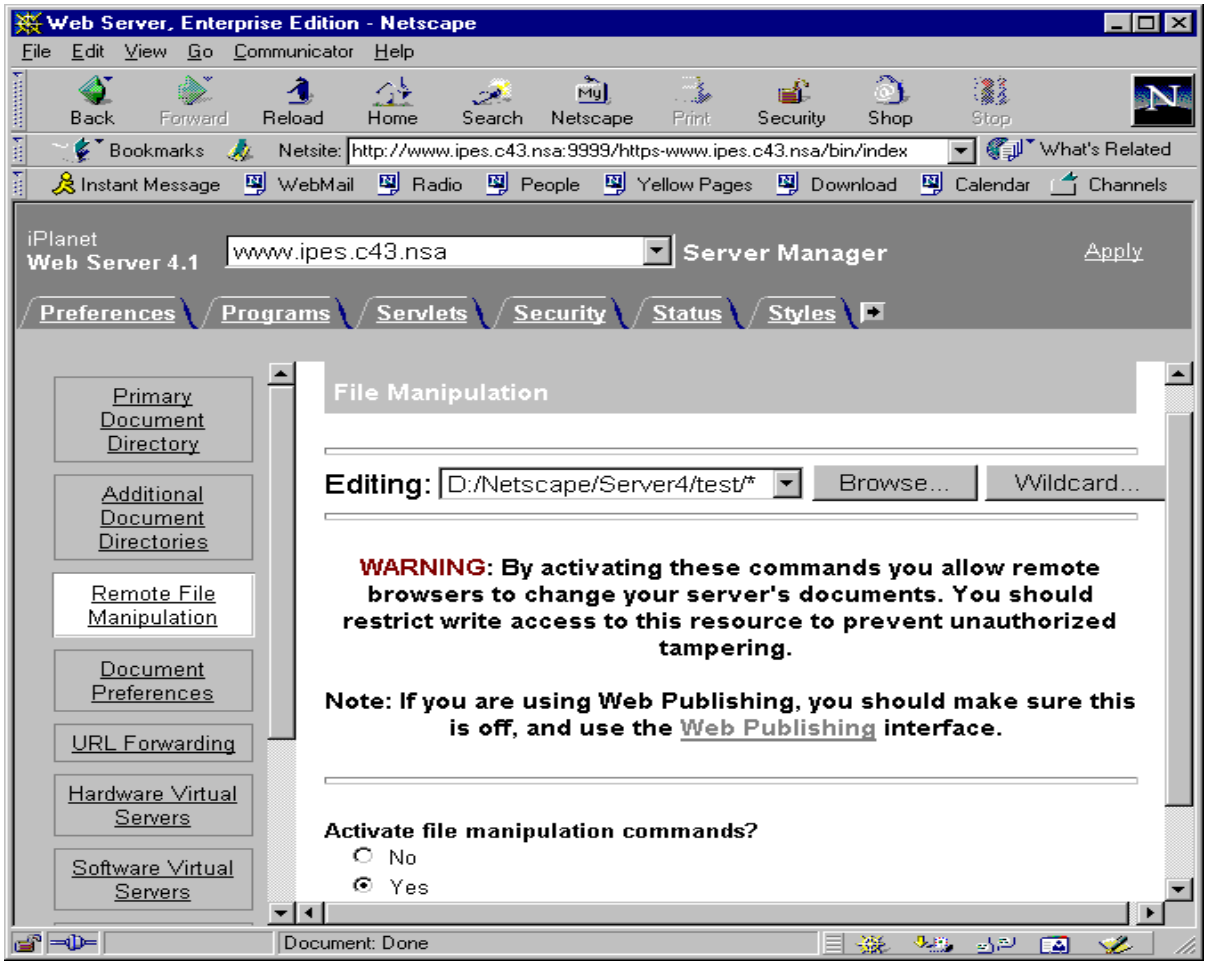

Figure 3-5 Remote File Manipulation Option in iWS.

Cautionary Note on Hardware and Software Virtual Servers

Hardware Virtual Servers (HVS) and Software Virtual Servers (SVS) are another option supported by iWS that provides flexible content management on the server. They are configured and managed under a server instance via Server Manager's Content Management tab. HVS permits multiple IP addresses on the server to be mapped to multiple document roots. Software Virtual Servers (SVS) offer the administrator the flexibility to support several web sites that are mapped to the same IP address on an existing server.

HVSs and SVSs share the same server configuration information as their parent server instance. For example, if SSL is enabled for one server, all servers will use SSL by default—to include one server certificate for all servers. This may lead to a naming conflict when a client receives the Server's Cert. For example, if the server instance www.organization.com server certificate's common name (CN) contains the name www.organization.com and a HVS or SVS for the server instance is named web1.organization.com, then a name conflict will arise when a client makes a SSL connection to web1.

It would be bad practice to encourage users to accept certificates that do not resolve the certificate's CN correctly, i.e., it could lead to a man-in-the-middle attack. If a site must use SSL and HVSs or SVSs, it would be preferable to have a certificate issued with the CN that could be used for both the server instance and its related HVSs and/or SVSs. In the example presented, *.organization.com could be used or (www|web1).organization.com; however, the certificate and private key should be placed on only one host and not shared amongst other hosts or server instances on the same host. In addition, the CA must be able to support this format. Additional information on this subject can be found at http://help.netscape.com/kb/corporate/19970210-11.html.

Parse HTML

Parse HTML allows the server to recognize and execute parsing commands within a static HTML document. This capability is sometimes called Server Side Includes (SSI) because it permits the server to include additional functionality in an HTML document. As shown in Figure 3-6, there are three settings to determine how the Parse HTML option will function. These settings will affect the security of iWS.

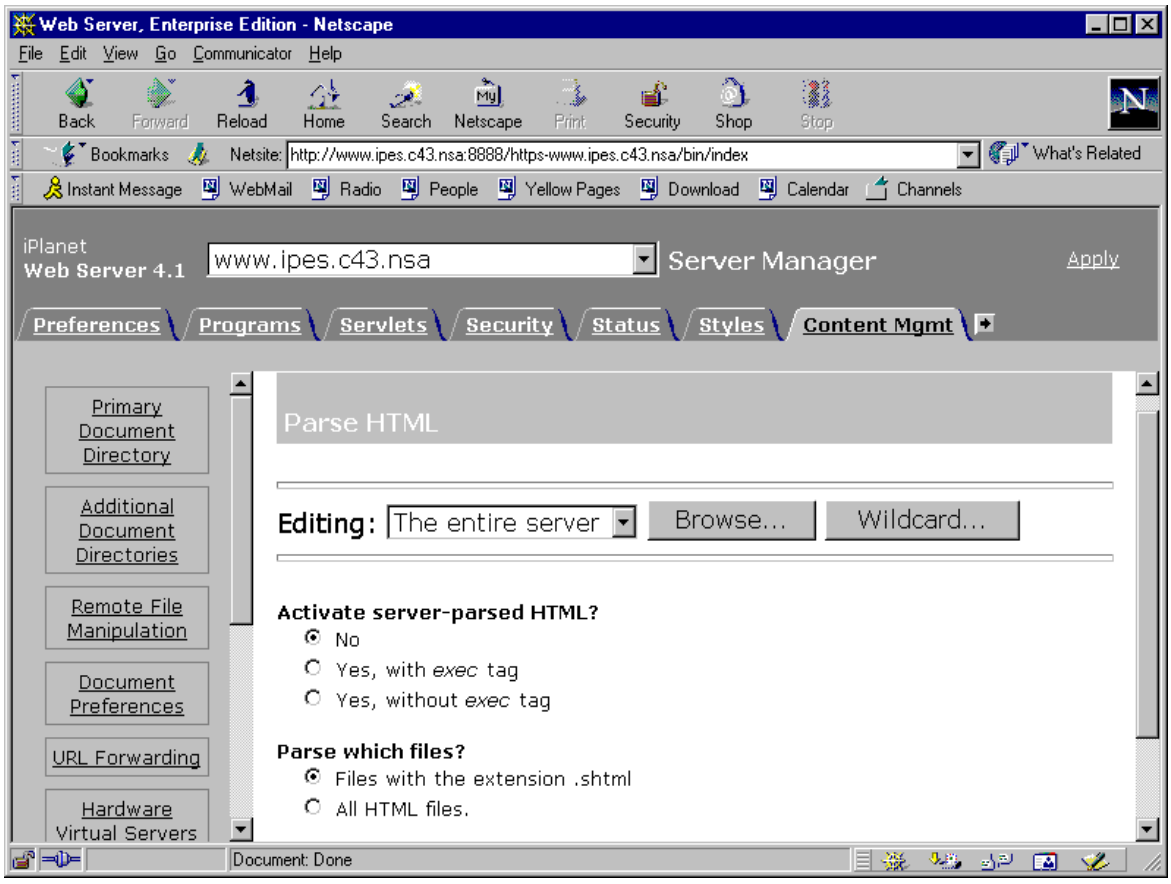

Figure 3-6 Parse HTML Option in iPlanet Web Server.

The Editing setting allows the administrator to select the file or folder where parsing will be permitted on the server. The other two settings can be appropriately set as they relate to that file or folder. For example, if a folder on the server has been established specifically for parsed HTML then server-parsed HTML documents in that folder can be set to function with or without the exec tag. In addition, further control based on the type

of file may be specified as either .shtml or all HTML documents. Thus, Editing allows the administrator to control with finer granularity the files that include server-parsed HTML.

The use of the exec tag is not recommended. If it is used, then additional precautions are necessary. This tag allows an HTML document to call an external, executable script, which will return its results to the calling HTML document. The flexibility to call programs residing on the server provides the opportunity for abuse. Therefore, the same web server security precautions used with cgi-bin programs must be applied when SSI with the exec tag is enabled.

ACLs may be used in conjunction with parsed HTML. A server-parsed HTML document under the control of an ACL may require user authentication and a valid IP address or host name. An ACL should be specified for an HTML document and not the program file called from the HTML document.

Cache Control Directives

Cache Control Directives is an option provided by iWS to control caching which occurs on an associated proxy server. iWS provides various settings to control the level of caching such as public, private, no cache, and maximum age of cached information. It is recommended that when sensitive information is being served or SSL sessions are enabled that this option be set to No Cache as shown in Figure 3-7. This will prevent any sensitive information from being left in the proxy server cache.

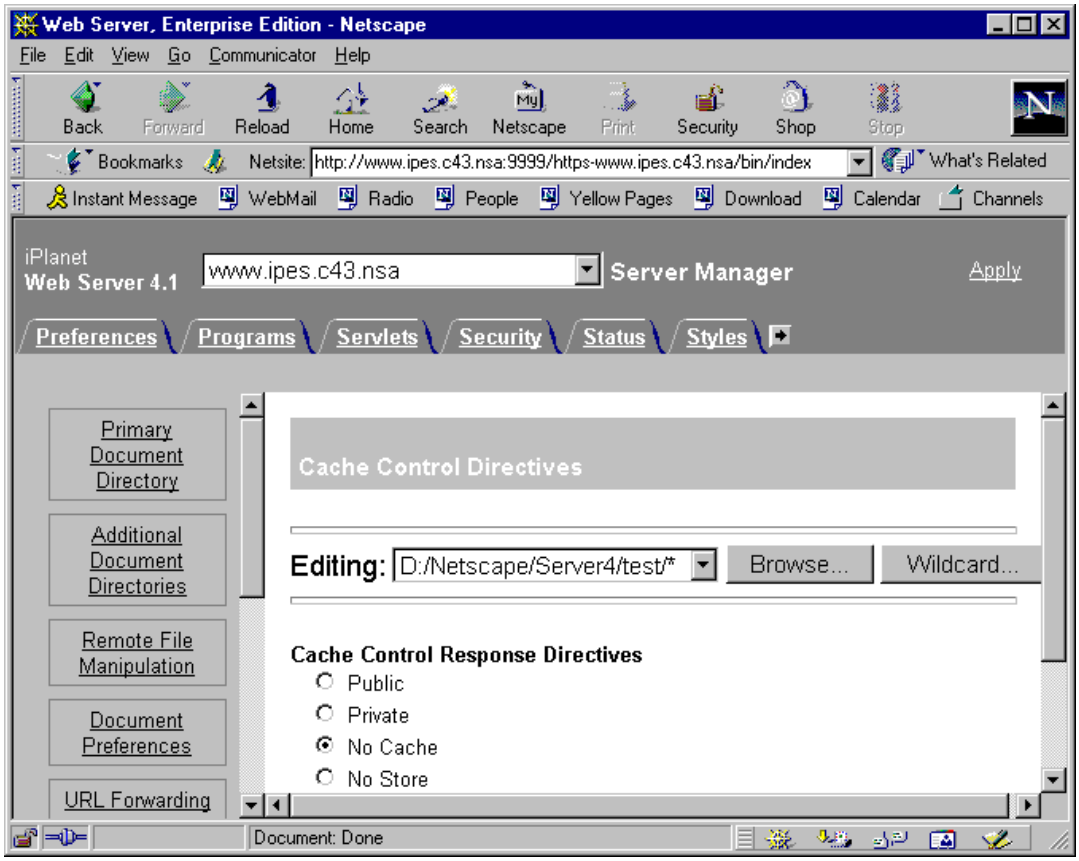

Figure 3-7 Cache Control Directives in iPlanet Web Server

Web Publishing

iWS includes a remote web publishing tool called Web Publisher. It runs as a Java applet in a remote user's web browser. Web Publisher supports file editing and manipulation as well as specific publishing functionality.

For example, after an HTML file has been modified and saved, it is re-published on the server by selecting the Publish Edited File option from the File submenu. The updated file will be saved back to the server and accessible to all users who have the appropriate access rights. Other functionality includes ACL and file locking.

The Web Publisher applet performs its functions by remotely sending and receiving commands and data over the network using HTTP 1.1. If the remote connection with the server has not been established securely using encryption, iWS information including user name and password is readable using a network sniffer. HTTP 1.1 uses base64 encoding to hide the user name and password. It is recommended that Web Publishing operations be performed only over secure links; however, it is important to note that Web Publishing is not supported when SSL is turned on. It is recommend that content management be performed using Remote File Manipulation over SSL or Web Publishing with some other form of secure communications, i.e., virtual private network technology such as Internet Protocol Security (IPSec) Protocol.

Web Publishing State

The Web Publishing State toggle button can be set to on or off. When toggled on, iWS allows remote users to use Web Publisher as described above. If the toggle button is off then iWS blocks all remote access attempts.

In order for a remote user to access the web publishing capabilities supported by Web Publisher, the iWS administrator must create a netshare user folder. By default, the folder and its contents are restricted to its owner. However, other users' folders may be accessed if appropriate rights have been granted. Any file manipulation requires user authentication; however, this authentication may not be protected if not performed over a secure channel. In addition, it is important to remember that Web Publishing could create a potential issue relating to access controls. A netshare folder owner has the ability to grant users access through the Web Publishing feature. If a netshare folder owner has granted specific users and/or groups access through the web Publishing feature, then file modification and manipulation will only be restricted as defined in the ACL.

The ACL provides varying levels of security for remote publishing capabilities in the Web Publisher applet. Therefore, careful consideration must be given to whom and to what extent access will be granted. If the Web Publishing State is to be toggled on then it is recommended that appropriate iWS ACLs be utilized on the netshare user folders. In addition, file permissions from within Windows NT should be applied to further restrict Web Publisher functions. Otherwise, the Web Publishing State should be disabled to avoid version control problems and possible malicious activities.

Unlock File

The Unlock File feature available under Web Publishing of iWS allows the administrator to override file locking on a file. File locking is used as a means of version control of

shared files on the server. Since Web Publisher permits multiple remote users to modify a file, version control is critical in order to track each user's modifications.

A typical sequence of file operations may consist of locking the file on the server, modifying the file locally, uploading the updated file to the server and releasing the lock when complete. Although one user may lock a file, the administrator may unlock it for another user by using the Unlock File feature. This feature poses an internal security concern. Any file could be unlocked without the initial locker's awareness and be modified or deleted by another user. The user who locked the file will not be notified of the modification or deletion. The Unlock File feature should be reserved as only a last resort to unlocking files for user access. It is recommended that file locking and unlocking be performed through the Web Publisher applet to provide the best version control.

Miscellaneous

Lightweight Directory Access Protocol Server and iWS

iWS was designed to be used with the iPlanet Directory Server (iDS). A directory server is used to manage users and groups. It should be noted that the iWS FAQ states "…other LDAP servers will work for authentication, but some features of iPlanet Directory Server such as Dynamic Groups and Password Policies will not be available." It is therefore recommended that only the iPlanet Directory Server or a similar directory server that can enforce a password policy be used. Lastly, directory server access logs should be monitored for BIND failures. In iDS, look for error codes of err=49. This is an indication of possible failed attempts of unauthorized access to iWS.

ACLCacheLifetime Directive

The ACLCacheLifetime directive, which can be manually set in the magnus.conf configuration file, determines how long a user's ACL credentials are valid. The file can be found in the *server_root/server_name/config/*. It is recommend that this setting be no greater than the default value of 120 seconds. If an administrator determines that the time should be greater, then the ACLCacheLifetime should be less than the directory server's "Password Lockout Reset Failure Count" and "Lockout Duration" times.

Directory Browsing

iWS does not have a feature to directly disable directory browsing. In order to do this in iWS, you should place an index.html or home.html file in each of the directories that you do not wish to be browsed. These files will serve as a default web page for the associated directory.

Robot Exclusion Protocol

A little known protocol known as the Robot Exclusion Protocol can be used to limit the scope of a web site indexing agent. Robots, spiders, crawlers, wanders, and bots are all names associated with web site indexing agents. These agents essentially go through all of the accessible links of a web site and index the associated web pages. This allows your web site to be searchable. An administrator can limit an "ethical" robot's indexing scope by including a robot.txt file in the root of the web site. It may be possible to try and

limit the activity of an "unethical" robot by identifying its hostname or IP through log activity and placing this information in an ACL.

The robot.txt file simply indicates to a robot what directories should and should not be indexed. An example that would exclude all robots from all directories is

User-agent: *
Disallow: /

For additional information on using robot.txt see *Web Servers, Security, & Maintenance* or visit the web site: http://info.webcrawler.com/mak/projects/robots/robots.html.

SPAMbots

SPAMbots are a form of robots that search a web site for e-mail addresses. They basically search the web site looking for MAILTO: tags. Organizations should carefully consider what e-mail addresses are necessary for publishing. When practical, use a general e-mail address for dealing with outside correspondence. If this is not possible, it may be preferable to implement a CGI mailer. For more information on SPAMbots, see *Web Servers, Security, & Maintenance*.

iWS Additional Security Considerations

The following additional security considerations are taken from the iWS Administrator's Guide.

- Limit Physical Access – Keep the server in a locked room that can be controlled so that only authorized administrators can enter.
- Limit Administration Access – Access to the Administration Server should be limited to as few individuals and computers as possible.
- Secure Your Key-Pair File – If backups of the key files are performed, it is important to safeguard these copies as well.
- Limit Other Applications - Careful consideration should be given to whether or not additional applications should be run on the server. The server could be subject to attack through bugs or holes in the application's code. In addition, consider what drives and directories that are shared and ensure that the appropriate file permissions are set.
- Prevent Clients from Caching SSL Files – Add the following statement inside the <HEAD> section of a file in HTML: <meta http-equiv="pragma" content="no-cache">.[12]
- Limit Ports – Disable any ports not used on the machine. Use routers or firewall configurations to prevent incoming connections to anything other than the absolute minimum set of ports.

[12] This does not prevent users from saving the file.

- SSL Limits – After receiving information from a client using SSL, consider how the information is handled and whether or not the information should be stored on the server.

Summary

iWS provides a complete enterprise solution offering several services to support Intranet and Internet requirements with integrated security features. This chapter described the security concerns of specific services and made recommendations on how to minimize the vulnerabilities that exist. In summary, the following guidelines should be applied:

- Use ACLs instead of Dynamic Configuration Files to control access.
- Use SSL 3.0 to protect sensitive content. Use strongest key possible.
- Use CGI directories instead of file types.
- Carefully review all programs utilizing the application interfaces supported in iWS. Review all Perl, Java, C++, JavaScript, and other program source code before installing it on the server to verify that it uses established programming practices. Do not put a script and script interpreter together in a directory for any reason. Utilize iWS ACLs and Windows NT file permissions to restrict write access to the directories containing these files.
- WAI is a powerful extension of iWS. The use of WAI applications should be carefully administered as follows:
    - Restrict remote user access to the iWS host machine.
    - Restrict write access to iWS folders where WAI applications reside.
    - Restrict WAI applications to run only on the iWS host machine.
- Enable SSL before using the Application Manager remotely.
- Actively monitor and maintain log files. Log files may help detect malicious activity and protect against future attacks.
- Recognize the potential security vulnerabilities of remote file manipulation and other content management tools such as Web Publisher. Restrict access using ACLs and user authentication where applicable.
- When using SSL with Virtual servers, ensure that no server name conflicts will occur in users' browsers.
- Use of the exec tag in SSI is not recommend.
- If using a proxy server, set the Cache Control Directive option appropriately if sensitive information is involved. "No Cache" is recommended.
- Web Publishing operations should be performed only over secure links; however, it is important to note that Web Publishing is not supported when SSL is turned on. It is

recommend that content management be performed using Remote File Manipulation over SSL or Web Publishing with some other form of secure communications, i.e., virtual private network technology such as Internet Protocol Security (IPSec) Protocol.

- The Unlock File feature should be reserved as only a last resort to unlocking files for user access.

- Use a directory server that can enforce a Password Policy.

- Ensure that the ACLCacheLifetime directive is not set to a time greater than the directory server's "Password Lockout Reset Failure Count" or "Lockout Duration."

- When required, use index files to prevent directory browsing.

- Limit "ethical" robots indexing scope by including a robot.txt file in the root of the web site.

- When practical, use a general e-mail address for dealing with outside correspondence or implement a CGI mailer.

- Implement the recommended additional security considerations in the iWS Administrator's Guide.

Chapter

4

Final Thoughts

This chapter provides additional information about applying settings and anti-virus programs. Anti-virus programs are an important part of the overall security of every information system and are strongly recommended for the iWS host systems.

Issues with Settings

It is important to note that when changing settings it is safer to first save the settings and then shut the respective server down and restart to implement the changes. In addition, subscribe to the news group secnews.netscape.com/netscape.server to keep up with current iWS issues.

Antiviral Program

Computer viruses are a major threat to all information systems. Some viruses may cause annoying consequences while others serious destruction. Regardless, all systems should have up-to-date anti-viral software installed to protect resources. It is also important to maintain the anti-viral program after installation by obtaining the most recent updates for protection against newly released viruses or strains of previously released viruses.

An excellent starting point for information and resources regarding viruses as well as other computer security resources is Purdue University's Computer Operations, Audit, and Security Technology (COAST) web site at http://www.cs.purdue.edu/coast/hotlist. COAST is now part of Purdue's Center for Education and Research in Information Assurance and Security (CERIAS) web site.

Appendix

A

References

[Cra] Richard Cravens, *Netscape Enterprise Server 3 Book*, Ventana Communications Group, 1997.

[Hay] James Hayes, "The Problem with Multiple Roots in Web Browsers – Certificate Masquerading," *Proceedings of the Seventh International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, IEEE, 1998.

[Hou] Russell Housley, Warwick Ford, Tim Polk, and David Solo, *RFC 2459 - Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, Network Working Group – Internet Engineering Task Force, 1999, http://www.ietf.cnri.reston.va.us/rfc/rfc2459.txt?number=2459.

[Hun] Jason Hunter and William Crawford, *Java Servlet Programming*, O'Reilly & Associates, Inc., 1998.

[IP1] iPlanet, *Administrator's Guide*, http://docs.iplanet.com/docs/manuals/enterprise/41/ag/contents.htm, 2000.

[IP2] iPlanet, *Netshare & Web Publisher Users Guide*, http://docs.iplanet.com/docs/manuals/enterprise/41/webpub/contents.htm.

[IP3] iPlanet, *Release Notes for iPlanet Web Server, Enterprise Edition Version 4.1 SP4*, http://docs.iplanet.com/docs/manuals/enterprise/41/rn41sp4.html.

[IP4] iPlanet, *Writing Web Applications with WAI: Netscape Enterprise Server/Fast Track Server, Version 3.0/3.01*, http://docs.iplanet.com/docs/manuals/enterprise/wai/index.htm.

[Lar] Eric Larson and Brian Stephens, *Administrating Web Servers, Security, & Maintenance*, Prentice Hall, 2000.

[NSA] National Security Agency, *Guide to Securing Microsoft Windows NT Networks*, 1999.

[Oak] Scott Oaks, *Java Security*, O'Reilly & Associates, Inc., 1998.

[Saf] *Security Alert for Enterprise Resources*, http://www.safermag.com/.

[Sch] Bruce Schneier, *Applied Cryptography, Second Edition,* John Wiley & Sons, Inc., 1996.

[Wag] David Wagner and Bruce Scheier, "Analysis of the SSL 3.0 Protocol," *The Second USENIX Workshop on Electronic Commerce Proceedings*, USENEX Press, November 1996, pp.29-40.

[Wya] Allen L Wyatt, *Netscape Enterprise Server,* Prima Publishing, 1996

Appendix

B

iWS Problems of Interest

Buffer Overflow in iPlanet Web Server 4, Server Side SHTML Parsing Module[13]

The vulnerability has been patched in iWS 4.1 SP4. The attack against previous versions of iWS can lead to a Denial-of-Service or remote execution of code in the context of the user that iWS is running as. 'Parsed HTML' option (server side parsing) must be enabled for vulnerability to be exploited.

The attack is carried out by sending a request of 198-240 characters (depending on the iWS version/platform) with extension .shtml (by default). iWS must have server side 'parsing' turned on. Overflow happens in logging function (when iWS tries to report that file is not found). If exploitation is successful (or iWS segfaults), nothing will remain in the logs.

Users of previous versions of iWS should upgrade to the latest versions. Until then, 'Parsed HTML' should be turned off. For more details review http://www.safermag.com/advisories/0010.html.

The remaining sections are excerpts from *Release Notes for iPlanet Web Server, Enterprise Edition Version 4.1 SP4*.

Upgrade Issues

Problem 388075. Upgrade to iPlanet Web Server 4.1 Does Not Update the obj.conf admin-check-admpw Setting

When you upgrade the iPlanet Web Server to version 4.1, the `admin-check-admpw` function in the Administration Server's `obj.conf` file includes the parameter `final=false`, when it should include `final=true`. This problem occurs only for upgrading the iPlanet Web Server, not for installing it.

Workaround

After upgrading, edit the `admin-check-admpw` function in the *`server_root`*`/https-admserv/config/obj.conf` file to change `final=false` to `final=true`.

Fixed Problems

- **382808.** LDAP over SSL breaks Manage User functionality

[13] S.A.F.E.R. Security Bulletin 001026.EXP.1.8, http://www.safermag.com/advisories/0010.html.

- **349465.** Install Certificate does not check key/cert match
- **495621.** Random crashes when using certificate based authentication
- **389276.** Adding server certificates to hardware token not working properly. Though the bug has been fixed, you still need to perform some steps to successfully add the server certificates.
- **394323.** SSL memory leak with client certificates.

Known Problems and Solutions

Problem 387590 (Linux, Solaris and NT platforms only). Encryption Ciphers/Settings Reset to Default Values

Changes you make to the cipher settings (in the Encryption Preferences page of the Administration Server interface) are not saved when you select the OK button and then restart the server.

Workaround

Change the cipher settings by editing the magnus.conf file. For more information, see the Security section of the magnus.conf appendix in the *NSAPI Programmer's Guide*.

Problem 400714. Web Publisher Applet and SSL

The Web Publisher Java applet is not supported in SSL mode.

Problem 383530 (Windows NT only). Encryption Error

The following error message may appear when you turn Encryption on with the default settings (in the Encryption On/Off page on the Preferences tab of either the Administration Server or the Server Manager), select Save and Apply, and enter your password:

Commit.exe Application error
the instruction at "0x780001146" referenced memory at "0x00661006d".
The memory could not be "written".
Click OK to terminate the application.

Select OK, and the following message appears:

Success! Your changes have been saved.

After you see this message, you should verify that the server has been restarted by going to the Shut Down or Server On/Off page on the Preferences tab or to the Services Control Panel.

Problem 396337 (Windows NT only). Configuration Change Windows Do Not Check for Correct Password (when SSL is Enabled)

If SSL is enabled on the server and you change any configuration on Windows NT platforms, and then click on "save and apply", the Web Server displays the "System error: Password did not match" error message.

Workaround

Click "Save" and then click "Apply" buttons at the right top of the Administration Server.