Global Information Assurance Certification Paper

Securing Microsoft Outlook Web Access server

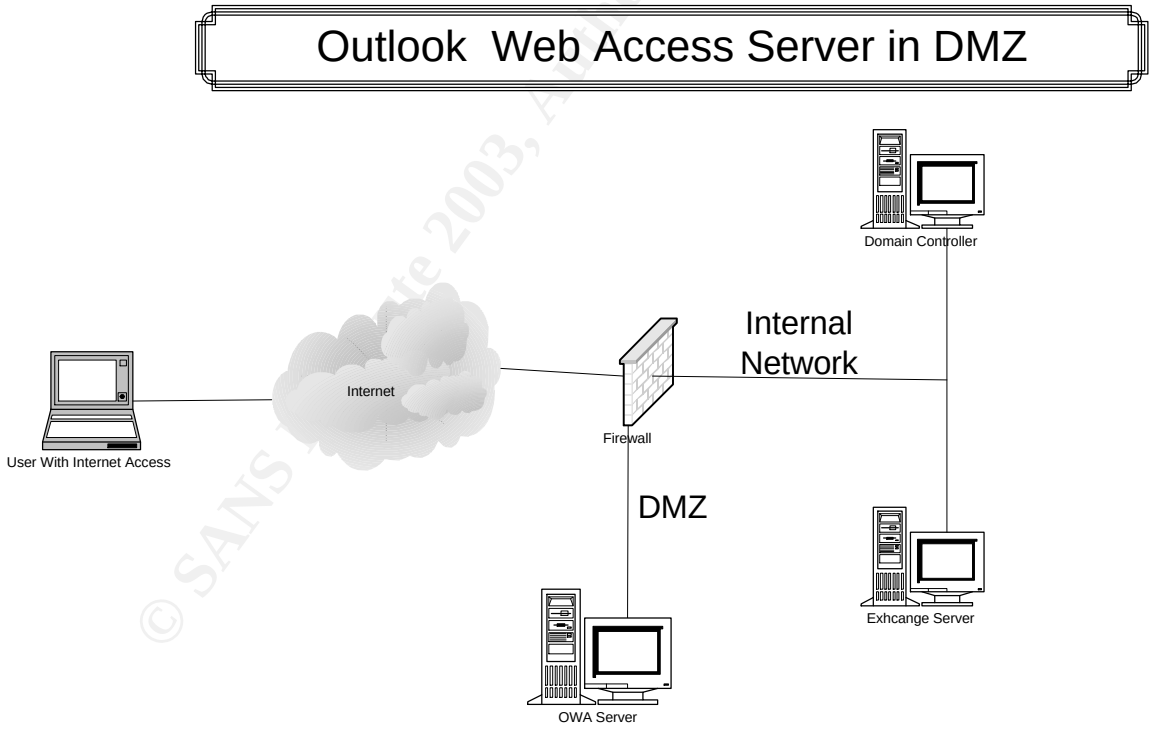
Practical for GSEC v1.4b Option 1

Grant Pederson
11/05/2002

ABSTRACT

In today's global environment, connectivity and convenience are quickly becoming commodities of corporate productivity. The ability to access either part or all of the corporate network from virtually anywhere has become reality. While the users reap the benefits of anytime, anywhere connectivity, the systems administrator in charge of securing all of these new avenues of entrance struggles to maintain tight network security. Microsoft Outlook Web Access (OWA) has provided a way for users to access email from any internet connected computer. Because one of the most widely deployed OWA configurations consists of an OWA server sitting in the DMZ or service network with connectivity to the internal mail and domain controller servers it becomes critically important to maintain the integrity of the OWA server.

This paper will present the secure implementation of Outlook Web Access from installation of the operating system to making the necessary firewall configurations. It will be assumed that the OWA server resides in a DMZ network and communicates with an internal Exchange 5.5 server and a Windows NT 4.0 Domain controller.



One might ask the question, "Why is securing the OWA server so important?" It is not only important, but critical for a couple reasons. First a compromised OWA server can provide information about the company's organizational structure that would be beneficial in helping any social engineering attempts. The global address book, while seemingly useless could prove to be

valuable information in the hands of a skilled or bold social engineer. With a complete list of company email addresses it is only a matter of time before an attacker finds someone to open a file containing a virus or backdoor Trojan. Names can also be used to deceive employees into doing something they would normally not do. For example, a hacker could call the helpdesk and pose as a senior official of the company, state that he/she changed their password last night and had already forgotten it, then request that it be changed.

A compromised OWA server can also serve as an origination point for further attacks. If set up correctly the OWA server should have several ports opened up to the Domain Controller and Exchange server on the internal network. A compromised OWA server could be used to exploit vulnerabilities on either of these two machines via the opened up ports on the firewall. In addition the owned OWA server might be used to attack other servers in the DMZ. Because the OWA server likely sits on the same network as the company's web server or other internet facing systems, it can be used to scan or attack the other machines with no intervention from the firewall.

In this paper several layers of protection will be discussed, and it will be shown how defense in depth can be implemented in your OWA configuration. Some of the necessary steps in configuring a secure OWA server that will be discussed are: the installation and configuration of the operating system on which OWA will be run, installing and configuring IIS on the operating system, the configuration of the Microsoft Exchange Server, and the firewall configuration needed to allow each machine to communicate with the others.

Operating System Installation and Configuration

Any new implementation of OWA typically begins with a clean install of the operating system. This seemingly trivial step is often paid little attention, thus setting the stage for disaster. Below is a basic checklist of settings that should be configured on the Windows NT Server used for Outlook Web Access. This list was compiled from a more comprehensive checklist located at: http://www-tus.csx.cam.ac.uk/pc_support/WinNT/ntsecchk.html. The following list highlights some of the more important settings that should serve as a bare minimum for any install.

Physical Security Procedures

1. Apply the latest service pack and post service pack hot fixes to address known vulnerabilities in the Windows NT operating system.
2. In order to prevent unauthorized remote access to the registry, carry out the procedure in MS Knowledge base article Q153183: How to restrict access to NT registry from a remote computer.

3. To prevent unauthorized users from seeing account policies etc. Carry out the procedure in MS Knowledge base article Q143474: How to restrict Information available to anonymous Logon Users.
4. To defeat dictionary attack and make brute force attacks more difficult: Carry out the procedure in MS Knowledge base article Q161990 : How to Enable strong password functionality in Windows NT
5. Remove the OS/2 and POSIX subsystems to reduce the number of exploitable processes. More details on this in the Hardening IIS section.
6. Check the running services and only run the ones that absolutely required or run under a less privileged account than the "system" account to reduce the number of exploitable services.

Account Related Procedures

1. Rename the Administrator account to something obscure to provide an additional barrier to a known privileged account.
2. Make administrative passwords extremely difficult and maximum length to defeat dictionary attacks and make brute force cracking difficult.
3. Create a new account named Administrator to serve as a decoy. Disable the account, and make sure to audit this account to detect any malicious logon attempts.
4. Disable the Guest account. Prevents the use of one of the built in accounts by which an attacker may try to gain access.
5. Set Maximum Password age (35) to force password changes.
6. Set Minimum Password age (14). This prevents users from circumventing the password uniqueness requirement by quickly changing their password.
7. Set Minimum Password length (8)
8. Set Password Uniqueness (3) to disallow the same password to be used within about a 3 month period.
9. Enable account lockout after 3-4 to prevent high speed dictionary/brute force attacks
10. Reset count after (25) minutes. Also helps to thwart automated attacks
11. Set lockout duration to at least 30 minutes but preferably forever.

User Rights Policies

1. Remove the Guest account from Access this computer from the Network to prevent hack attempts using the guest account
2. Set log on locally to only people that will need access to the OWA server for email. For this setup this will include the IUSER_servername account in addition to the domain users that will need email access.
3. Set auditing policies, and audit Logon and Logoff, and File and Object Access

IIS procedures

1. Install only minimal NT and internet services. Do not install or enable services you do not need including the FTP, SMTP, and NNTP Servers included in IIS.
2. Set appropriate virtual directory permissions.
3. Do not accept defaults for data directory: Store the web site on a different volume from the OS. This makes it more difficult for an intruder to find files, and helps thwart some worms that attempt to access default directories.

Harden IIS

After installing the base operating system, you will need to install IIS. IIS will be used to interface with the user requesting email from the internet. IIS is installed under Start>Settings>Control Panel>Add/Remove Programs>Add/Remove Windows components. After installing IIS you will need to harden the application much the same as we did with the operating system.

A default installation of IIS is as insecure as a default installation of Windows NT. For this reason we must take additional steps to harden the IIS Application. The first step in this process is to install all of the latest security hot fixes. In order to determine what patches your system is lacking it is suggested that you learn to use the HFnetchk tool. The HFnetchk tool can be downloaded at : http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q315665& The HFNetchk tool will scan your system for missing service packs and hot fixexs.

Using HFnetchk is fairly straightforward. To scan the local computer simply navigate to the directory in which you placed the hfnetchk.exe downloaded from the above address and type hfnetchk -? to see all of your options. By typing hfnetchk –v the local machine will be scanned and the output will be displayed in verbose mode. All missing hotfixes will be displayed along with their corresponding Microsoft article numbers so you can easily find them on the Microsoft web site.

Next you will want to remove the OS/2 and POSIX subsystems. These are leftovers from the old 3.x days that allowed NT to run code from other operating systems. This was a good selling point back then given NT's weak market share, but these subsystems are very rarely used today and only introduce more weaknesses to the IIS application. To remove these subsystems you will need to edit the registry. Below are the registry modifications that must be made to eliminate these subsystems.

Hive	HKEY_LOCAL_MACHINE\SOFTWARE
Key	\Microsoft\OS/2 Subsystem for NT
Action	Delete all sub keys

Hive	HKEY_LOCAL_MACHINE\SYSTEM
Key	\CurrentControlSet\Control\Session Manager\Environment
Name	Os2LibPath
Action	Delete Os2LibPath key

Hive	HKEY_LOCAL_MACHINE\SYSTEM
Key	\CurrentControlSet\Control\Session Manager\SubSystems
Action	Delete Optional, Posix and OS/2 keys

Then delete the winnt\system32\os2 directory and all subdirectories and reboot the machine.

Move and ACL Critical files

Place all commonly used administrative tools in a special directory out of %systemroot% and ACL them so that only administrators have full access to these files. For example create a directory called \CommonTools and place the following files in there:

xcopy.exe	wscript.exe	cscript.exe	net.exe	ftp.exe	telnet.exe
arp.exe	edlin.exe	ping.exe	route.exe	at.exe	finger.exe
posix.exe	rsh.exe	atsvc.exe	qbasic.exe	runonce.exe	syskey.exe
cacls.exe	ipconfig.exe	rcp.exe	secfixup.exe	nbtstat.exe	rdisk.exe
debug.exe	regedt32.exe	regedit.exe	edit.com	netstat.exe	tracert.exe
nslookup.exe	rexec.exe	cmd.exe			

This will make it much more difficult for any unauthorized use of these system files by an intruder.

Disable or Remove all Sample applications

Normally you would want to remove all sample applications and directories including:

?\inetpub\iissamples
?\inetpub\iissamples\sdk
?\inetpub\AdminScripts
?\Program Files\Common Files\System\msadc\Samples

These particular directories contain sample scripts and applications that are not to be installed on a production server by any means. You can either remove these directories manually or allow IIS Lockdown to do it for you. IIS Lockdown is explained in detail in the following sections.

Enabling Auditing

Enabling auditing is extremely important so that you can easily identify unauthorized activity on your OWA server. In the event your server is compromised detailed logging can also provide the evidence required to pursue legal action against the perpetrator. It is recommended that you use the W3C Extended logging format by following this procedure:

1. Load the Internet Information Services tool.
2. Right click on the OWA website and click on properties.
3. Click on the Web Site tab.
4. Check the enable logging checkbox, and choose the W3C Extended log file format from the drop down list and select at a minimum the following properties:
   *Client IP Address
   *User Name
   *Method
   *URI Stream
   *HTTP Status
   *Win32 Error
   *User Agent
   *Server IP Address
   *Server Port

As always you may select to audit in greater detail depending on your personal preference. In this author's opinion it is always best to err on the side of over-logging. While this may make the log files larger it provides more detail as to what is happening on your server.

Disable RDS Support

Remote Data Service (RDS) is a component of Microsoft Data Access Components (MDAC) installed by default with IIS. The goal of the RDS component is to enable controlled Internet access to remote data resources through the Windows NT's IIS. However, because the RDS DataFactory (a single component of RDS) allows implicit remoting of data access requests by default, it can be exploited to allow unauthorized Internet clients to access OLE database (DB) datasources available to the server. The implicit remoting function of the RDS 1.5 through the DataFactory component should be disabled. To disable RDS support you will need to delete the following registry keys.

- **HKEY_LOCAL_MACHINE \SYSTEM \CurrentControlSet \Services \W3SVC \Parameters \ADCLaunch \RDSServer.DataFactory**
- **HKEY_LOCAL_MACHINE \SYSTEM \CurrentControlSet \Services \W3SVC \Parameters \ADCLaunch \AdvancedDataFactory**

- **HKEY_LOCAL_MACHINE \SYSTEM \CurrentControlSet \Services \W3SVC \Parameters \ADCLaunch \VbBusObj.VbBusObjCls**

Additional information on RDS support can be located at Microsoft Knowledge Base article MS98-004.

Once we have brought IIS up-to-date with the latest hot fix release, and have taken additional steps to harden the application we can proceed to installing the Outlook Web Access components and locking down the IIS application with IIS lockdown.

Installing OWA Components

Once IIS has been installed and properly configured it is time to install the Outlook Web Access Components from the Exchange Server CD. The following section will explain the installation and proper permissions that must be assigned to allow OWA to function securely.

To Install the OWA Components from the Exchange server disk simply run the setup.exe and choose to install OWA components only. Once the active server page components have been installed you will need to grant the 'Log on Locally' and 'Access this computer from Network' rights only to users that will need access to web based email. If the IIS computer is a member of a domain (which is typically the case) it is recommended you create a local group on the OWA server, add the domain users, group, or groups that will require access to this local group, then grant the local group the 'Log on Locally' right. The 'Access this computer from Network' right should be granted to the Authenticated Users group, which will include everyone with a valid domain Username and Password.

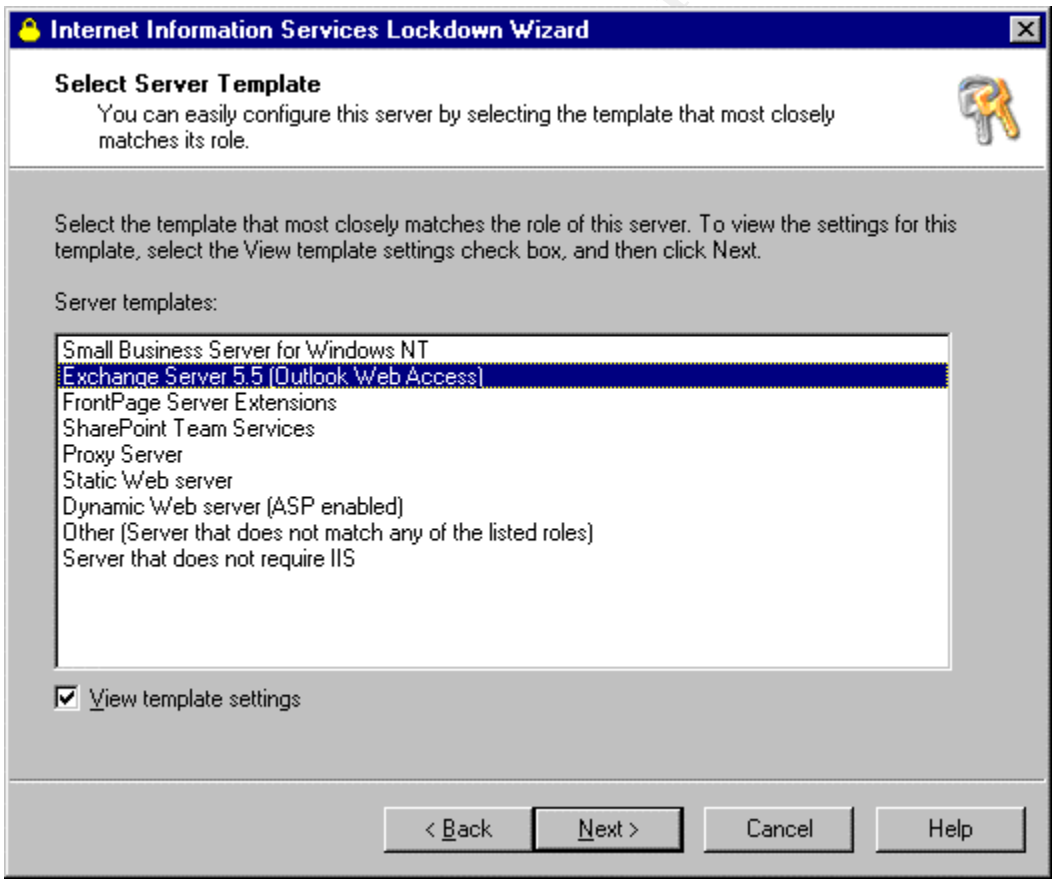
IIS Lockdown / URLScan

The default installation of IIS includes four different servers: NNTP, SMTP, WWW, and FTP. The only service required for OWA to function correctly is the WWW service. In the following example we will run IIS Lockdown to turn off the unneeded services in addition to making several other necessary security configurations. In this example we will:
Install URLScan
Disable unneeded script mappings
Remove IIS samples virtual directory
Remove MSADC virtual directory
Remove IIS admpwd virtual directory

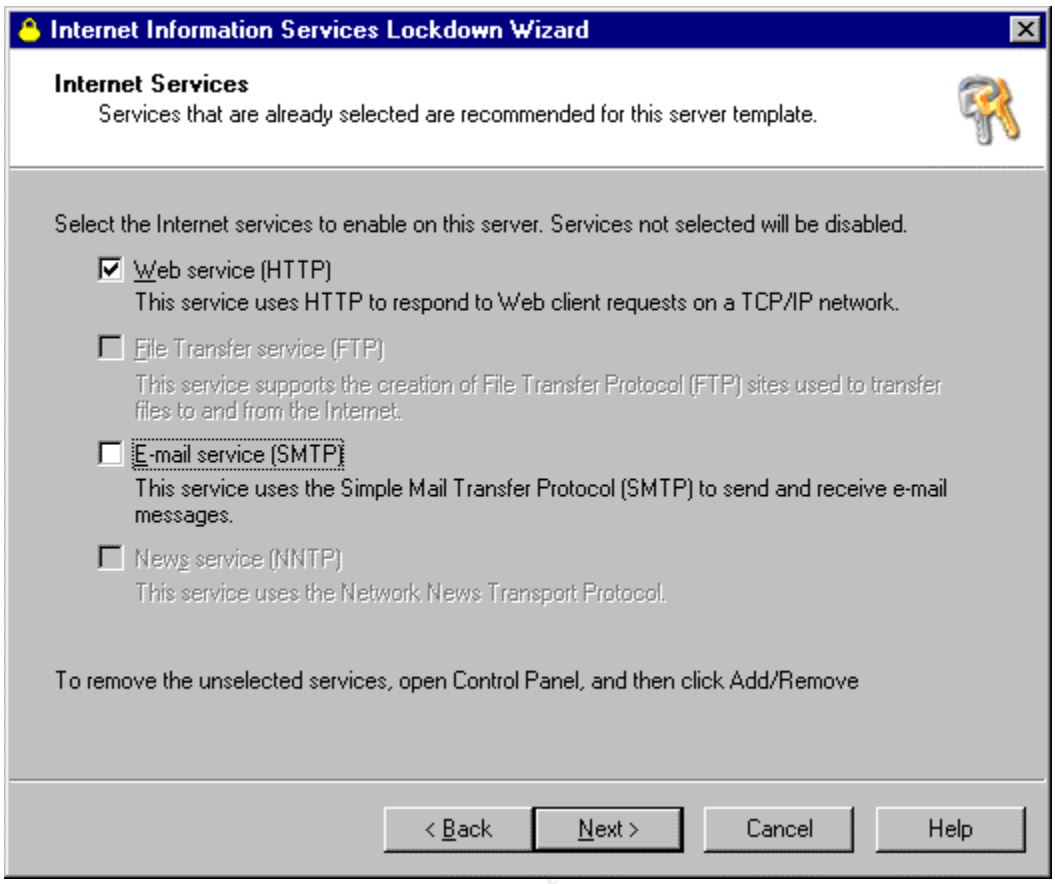
Remove Scripts virtual directory
Remove IIS Admin virtual directory
Remove IIS help virtual directory
Deny execute permission for system utilities to anonymous user account
Deny write permission to web content directories to anonymous user account

Begin by downloading IIS Lockdown from:
http://www.microsoft.com/downloads/release.asp?ReleaseID=43955&area=search&ordinal=1

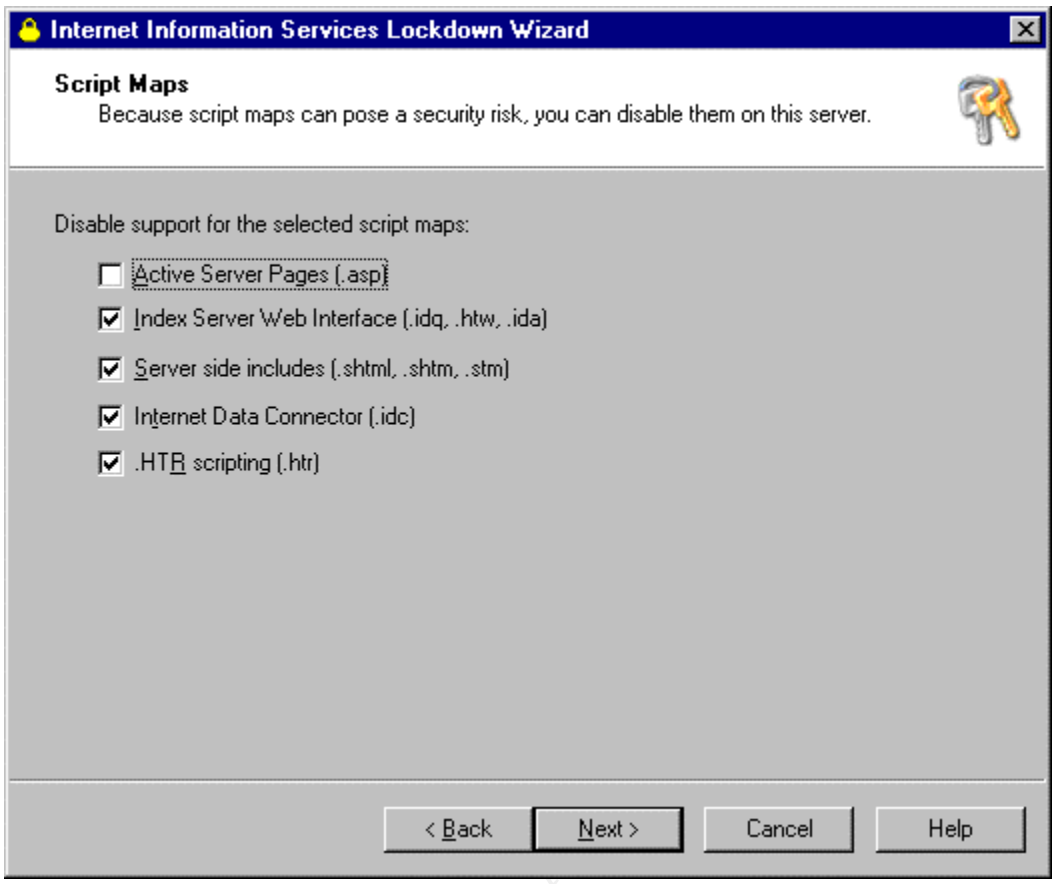
Continue the setup by running the iislockd.exe file. This will start the IIS Lockdown setup wizard. Click next at the initial screen.



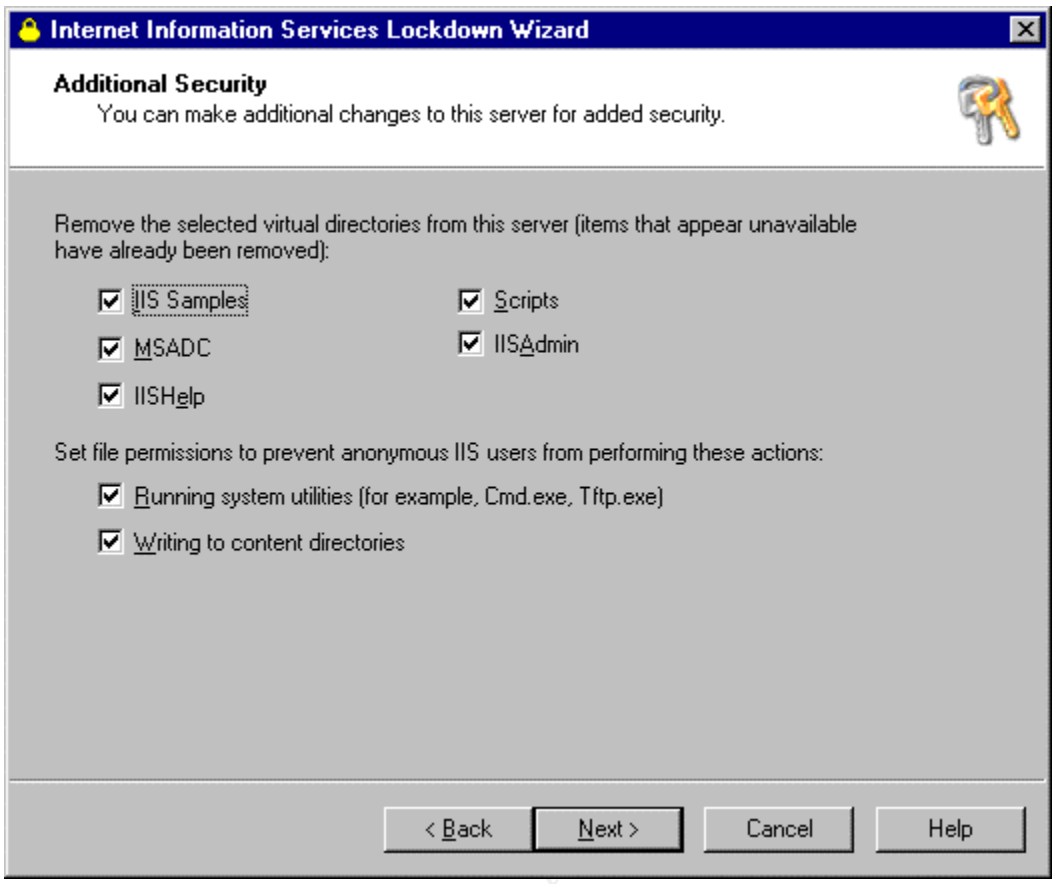
Select the Exchange Server template and check the View template settings checkbox. Then click next.



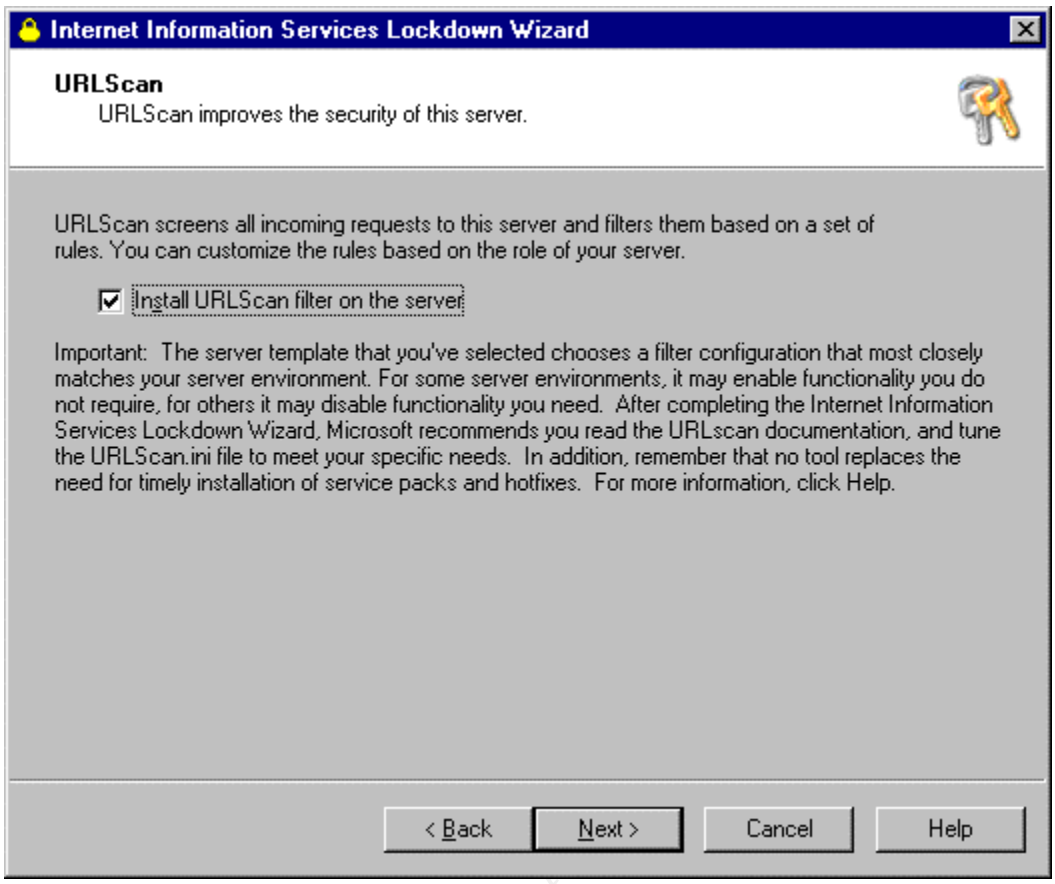
De-select the E-mail service (SMTP) checkbox and click next.



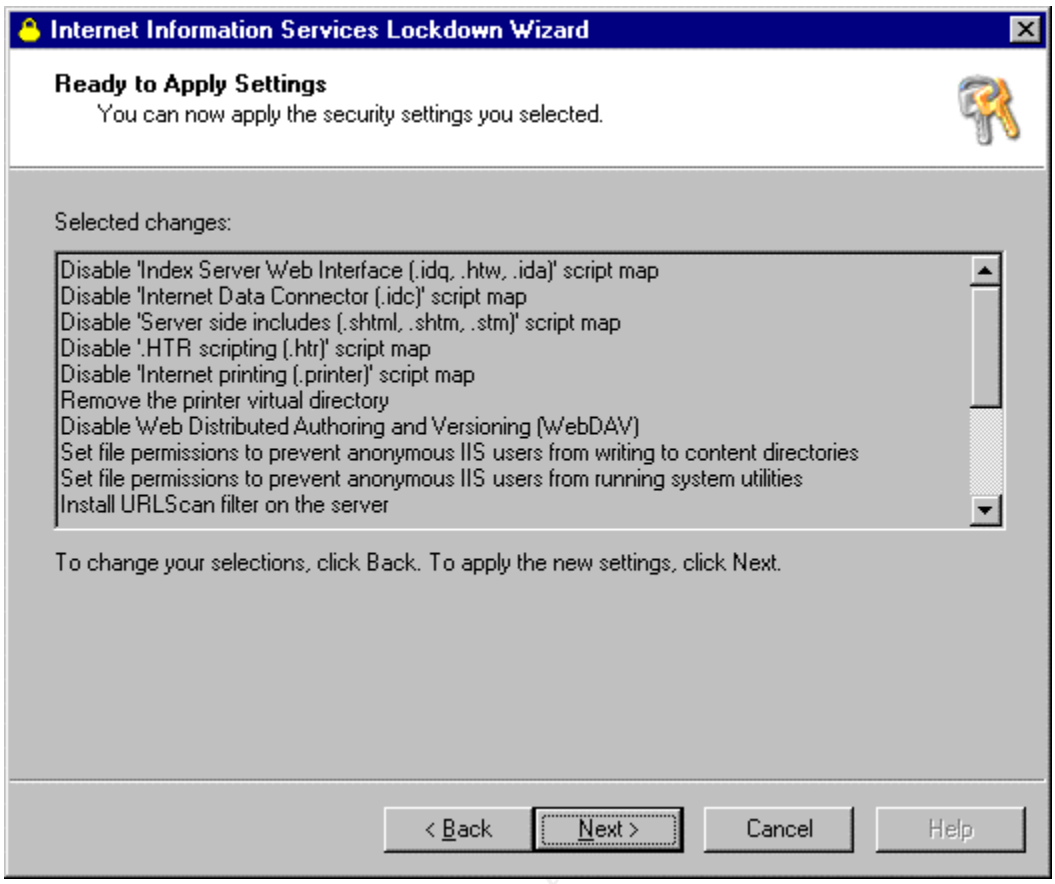
Make sure that the Active Server Pages (.asp) checkbox is not selected. OWA will not run without support for Active Server Pages. Select the check box next to all other options to disable support for the selected script maps.



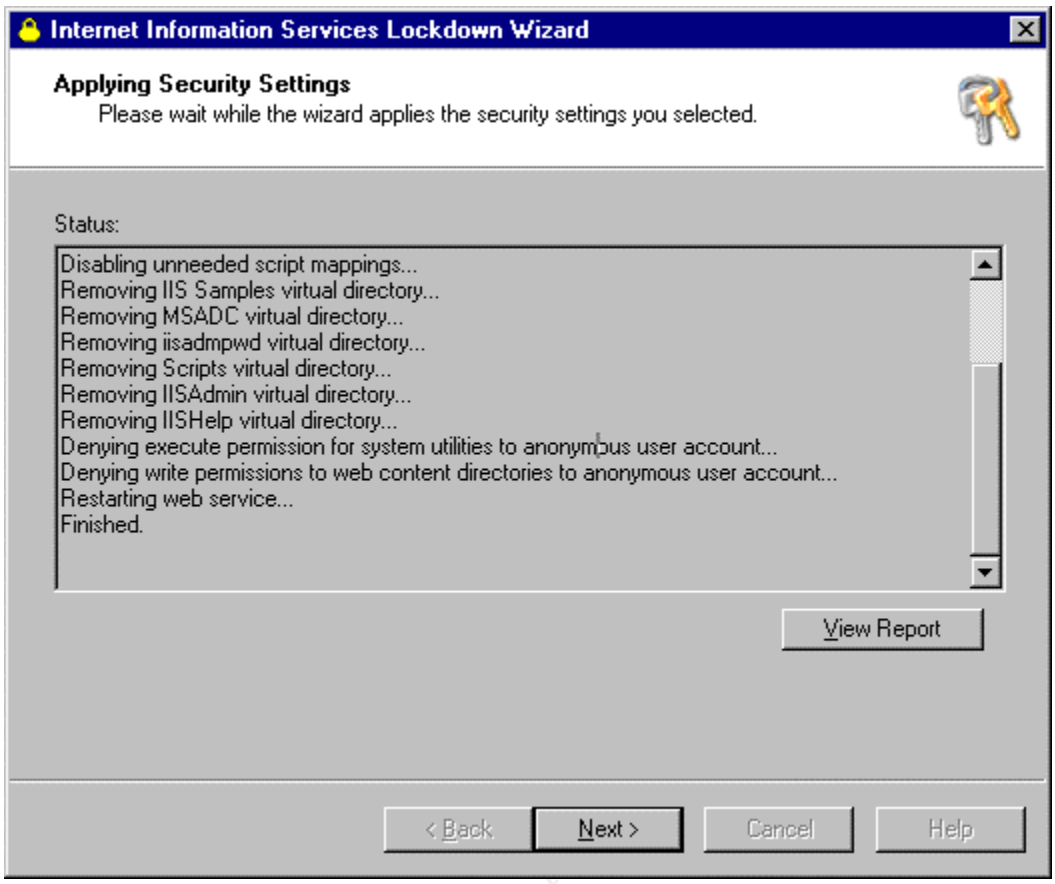
Make sure that all virtual directories are removed that are not needed by OWA, and that you choose to restrict file permissions for anonymous IIS users.



Check Install URLScan filter on the server, and click next.



Double check your settings and click next.



IISLockdown will then complete the required tasks, and the setup will be complete. The elimination of the directories listed above is important because they contain exploitable samples or provide other un-needed functionality. The IISAdmin virtual directory, for example, allows you to reset Windows NT passwords and is designed primarily for intranet scenarios. This directory should always be removed if the web server is connected to the internet. More information on this functionality can be found in Microsoft Knowledge Base article Q184619. The sample directories should never be installed on a production server because of the numerous exploits that rely on these sample directories and their sample applications being available. Removing them will eliminate any chance of their contents being used against you.

URLScan configuration

The URLScan ISAPI filter screens inbound requests to the web server. The filter uses rules configured by the administrator to block requests that contain potentially dangerous strings that could compromise a web server. If you installed the IIS Lockdown in the previous section URLScan should have been installed already. If not you can download URLScan from: http://www.microsoft.com/downloads/release.asp?ReleaseID=37756&area=search&ordinal=3

Once installed, URLScan must be configured to work with your specific application, in this case Outlook Web Access.

URLScan parameters are configured by modifying the **%windir%\System32\Inetsrv\URLscan\urlscan.ini file**. In its present state OWA will not function correctly so we will need to modify the urlscan.ini file for use with Outlook Web Access. Your version of IISLockdown may include a template for this file that will work with OWA, but if this is not the case you will have to manually configure the urlscan.ini file. Below is an example of a working configuration for the urlscan.ini file being used on an OWA server. The text below can be cut and pasted into the **%windir%\System32\Inetsrv\URLScan** directory and named **urlscan.ini.**

```
[options]
UseAllowVerbs=1             ; if 1, use [AllowVerbs] section, else use [DenyVerbs]
section
UseAllowExtensions=0        ; if 1, use [AllowExtensions] section, else use
[DenyExtensions] section
NormalizeUrlBeforeScan=1    ; if 1, canonicalize URL before processing
VerifyNormalization=1       ; if 1, canonicalize URL twice and reject request if a
change occurs
AllowHighBitCharacters=1    ; if 1, allow high bit (ie. UTF8 or MBCS) characters
in URL
AllowDotInPath=0            ; if 1, allow dots that are not file extensions
RemoveServerHeader=1        ; if 1, remove "Server" header from response
EnableLogging=1             ; if 1, log UrlScan activity
PerProcessLogging=0         ; if 1, the UrlScan.log filename will contain a PID
(ie. UrlScan.123.log)
AllowLateScanning=0         ; if 1, then UrlScan will load as a low priority filter.
PerDayLogging=1             ; if 1, UrlScan will produce a new log each day with
activity in the form UrlScan.010101.log
RejectResponseUrl=          ; UrlScan will send rejected requests to the URL
specified here. Default is /<Rejected-by-UrlScan>
UseFastPathReject=0         ; If 1, then UrlScan will not use the
RejectResponseUrl or allow IIS to log the request

; If RemoveServerHeader is 0, then AlternateServerName can be
; used to specify a replacement for IIS's built in 'Server' header
AlternateServerName=

[AllowVerbs]

;
; Note that these entries are effective if "UseAllowVerbs=1"
```

```
; is set in the [Options] section above.
;

GET
HEAD
POST

[DenyVerbs]

;
; Note that these entries are effective if "UseAllowVerbs=0"
; is set in the [Options] section above.
;

PROPFIND
PROPPATCH
MKCOL
DELETE
PUT
COPY
MOVE
LOCK
UNLOCK
OPTIONS
SEARCH

[DenyHeaders]

;
; Request headers listed in this section will cause UrlScan to
; reject any request in which they are present.
;
; Headers should be listed in the form
;   Header-Name:
;

Translate:
If:
Lock-Token:

[AllowExtensions]

;
; Extensions listed here are commonly used on a typical IIS server.
;
; Note that these entries are effective if "UseAllowExtensions=1"
```

```
; is set in the [Options] section above.
;

.asp
.cer
.cdx
.asa
.htm
.html
.txt
.jpg
.jpeg
.gif

[DenyExtensions]

;
; Extensions listed here either run code directly on the server,
; are processed as scripts, or are static files that are
; generally not intended to be served out.
;
; Note that these entries are effective if "UseAllowExtensions=0"
; is set in the [Options] section above.
;

; Deny executables that could run on the server
.exe
.bat
.cmd
.com

; Deny infrequently used scripts
.htw     ; Maps to webhits.dll, part of Index Server
.ida     ; Maps to idq.dll, part of Index Server
.idq     ; Maps to idq.dll, part of Index Server
.htr     ; Maps to ism.dll, a legacy administrative tool
.idc     ; Maps to httpodbc.dll, a legacy database access tool
.shtm    ; Maps to ssinc.dll, for Server Side Includes
.shtml   ; Maps to ssinc.dll, for Server Side Includes
.stm     ; Maps to ssinc.dll, for Server Side Includes
.printer ; Maps to msw3prt.dll, for Internet Printing Services

; Deny various static files
.ini     ; Configuration files
.log     ; Log files
.pol     ; Policy files
```

```
.dat     ; Configuration files

[DenyUrlSequences]
..  ; Don't allow directory traversals
./  ; Don't allow trailing dot on a directory name
\   ; Don't allow backslashes in URL
:   ; Don't allow alternate stream access
%   ; Don't allow escaping after normalization
&   ; Don't allow multiple CGI processes to run on a single request
```

Password Authentication Methods

While a few different password authentication methods exist, for the purposes of this document we will utilize the combination of SSL and Basic Authentication. For a discussion of other password authentication schemes refer to: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/exchange/exchange55/deploy/confeat/olewebac.asp To use basic authentication over SSL we must first create a new virtual directory on the IIS server that is running the corporate website. For our example I will create a new virtual directory called inbox on the testsite.com website. The URL that users will then need to type to access email will be www.testsite.com/inbox.

Within the /inbox directory you will need to create a simple re-direct that forwards all traffic to httpS://OWA_server_IP/ exchange (notice the S). Below is a sample statement that will accomplish this task:

```
<html>
<head>
<meta http-equiv =”Refresh” content = “0,URL=HTTPS://<OWA_IP>/exchange”>
</head>
<body>
</body>
</html>
```

This will force all traffic to the OWA server to occur over SSL. This is important because basic authentication occurs in plain text and without SSL to protect the user’s account information, logon data could easily be sniffed out.

Once you have configured the IIS server to direct all traffic to the OWA server over SSL you will then want to go to the properties of the exchange virtual directory on the OWA server and click on the Directory Security tab. From there you will need to click on the Edit button under the Secure Communications section and select “Require Secure Channel when accessing this resource”. This will only allow secure connections to be made to the exchange virtual directory thus enforcing our requirement of secure communications with the OWA server.

Now that we are passing our account information securely from the client making the request to the OWA Server, we must allow the OWA Server to communicate securely with the PDC and Exchange Servers. To do this, a couple of rules need to be configured on the firewall to permit the traffic.

Exchange Server Configuration

By default the OWA Server and Exchange server will pass directory service and information store data on randomly chosen ports. Obviously opening up all possible ports from your DMZ to the internal network is not an option. You will need to configure the exchange server to communicate directory service and information store data on a specified static port to reduce the "hole" that is required to be opened through the firewall to the internal network. To do this you will need to edit the registry of the Exchange Server. A detailed explanation of this process can be found at: http://support.microsoft.com/default.aspx?scid=kb;EN-US;q259240 Below is an overview of the Registry tweaks that need to be made.

1. Start Registry Editor
2. Locate:
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\MSExchangeDS\Parameters
3. Add the following entry for the Microsoft Exchange Directory service:
   Entry: TCP/IP port
   Type: REG_DWORD
   Data: *port number to assign*
   NOTE: Do not assign ports immediately above the 1023 range. This may cause other problems on the Exchange Server.
4. Add the following entry for the Exchange Server information store:
   Entry: TCP/IP port
   Type: REG_DWORD
   Data: *port number to assign*
   NOTE: Do not assign ports immediately above the 1023 range. This may cause other problems on the Exchange Server computer.
5. Quit the registry editor.
6. Reboot the Exchange server

Firewall Rules

Once you have configured the exchange server to communicate on the ports you specified, you must create firewall rules to allow communications both from the client to the OWA Server and also between the OWA Server and the

PDC/Exchange servers. Begin by creating a new rule that allows all IP addresses to communicate with the OWA Server over TCP port 443 (SSL). Next you will need to create a rule to allow TCP ports 139, 135, and the two TCP ports designated for the directory service and information store data in addition to UDP ports 138, and 139 to pass from the OWA Server to the Exchange Server and PDC. A similar rule must also be configured that allow the same ports for communication outbound from the PDC/Exchange servers to the OWA Server.

It is essential that you restrict the rule to the IP addresses of the Exchange Server/PDC and the OWA server. For example the rule will state that the above ports are only allowed inside to the PDC's IP address if it is coming from the OWA server's IP address. To further restrict the rule and prevent IP spoofing from tricking the rule into allowing illegitimate traffic in, you can define MAC addresses for the machines and limit the rule to the corresponding MAC addresses of the servers.

Conclusion

While the benefits of anytime anywhere email are obvious the associated risks of implementing such a system are often overlooked. Any system directly accessible from the internet is eventually going to attract un-authorized attention. It is critical that you are prepared for this, and have done everything possible to protect your company's precious information. A secure operating system with a properly configured IIS server installed in conjunction with good administration and monitoring can greatly reduce the risk associated with exposing your corporate data to the internet. By investing the time and effort to implement the above recommendations you will be better prepared to thwart the black hat community when they come knocking at the OWA server's door.

References

University of Cambridge Computing Service, Technical User Support, "Microsoft Windows NT Security Checklist" URL:
http://www-tus.csx.cam.ac.uk/pc_support/WinNT/ntsecchk.html

Tim Hatton, "[NTSEC] OWA Security model", Jan. 26, 2000, URL:
http://security-archive.merton.ox.ac.uk/nt-security-200001/0011.html

Michael Parker, "Securing Web Based Corporate E-Mail Using Microsoft Exchange Outlook Web Access" July 26, 2001, URL:
http://rr.sans.org/email/corp_email.php

Terri Carroll, “Basic IIS 5.0 Default Web Server Security” April 11, 2001, URL: http://rr.sans.org/web/IIS5_sec.php

Microsoft Documentation. “HOW TO: Use URLScan with Exchange Outlook Web Access in Exchange Server 5.5” URL: http://support.microsoft.com/default.aspx?scid=kb;en-us;Q313131

Microsoft Documentation, “XWEB: How to Configure OWA to Connect to Exchange Through a Firewall (Q259240)”, URL: http://support.microsoft.com/default.aspx?scid=kb;en-us;Q259240

Microsoft Documentation, “Planning and Deploying Microsoft Outlook Web Access Abstract:, URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/exchange/exchange55/deploy/confeat/olewebac.asp

Microsoft Documentation, “Troubleshooting Guide for Microsoft Outlook Web Access”, URL: http://support.microsoft.com/default.aspx?scid=/support/exchange/content/whitepapers/owa_tshoot.asp&FR=0

Microsoft Documentation, “Internet Information Server Baseline Security Checklist”, URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/iis4cl.asp