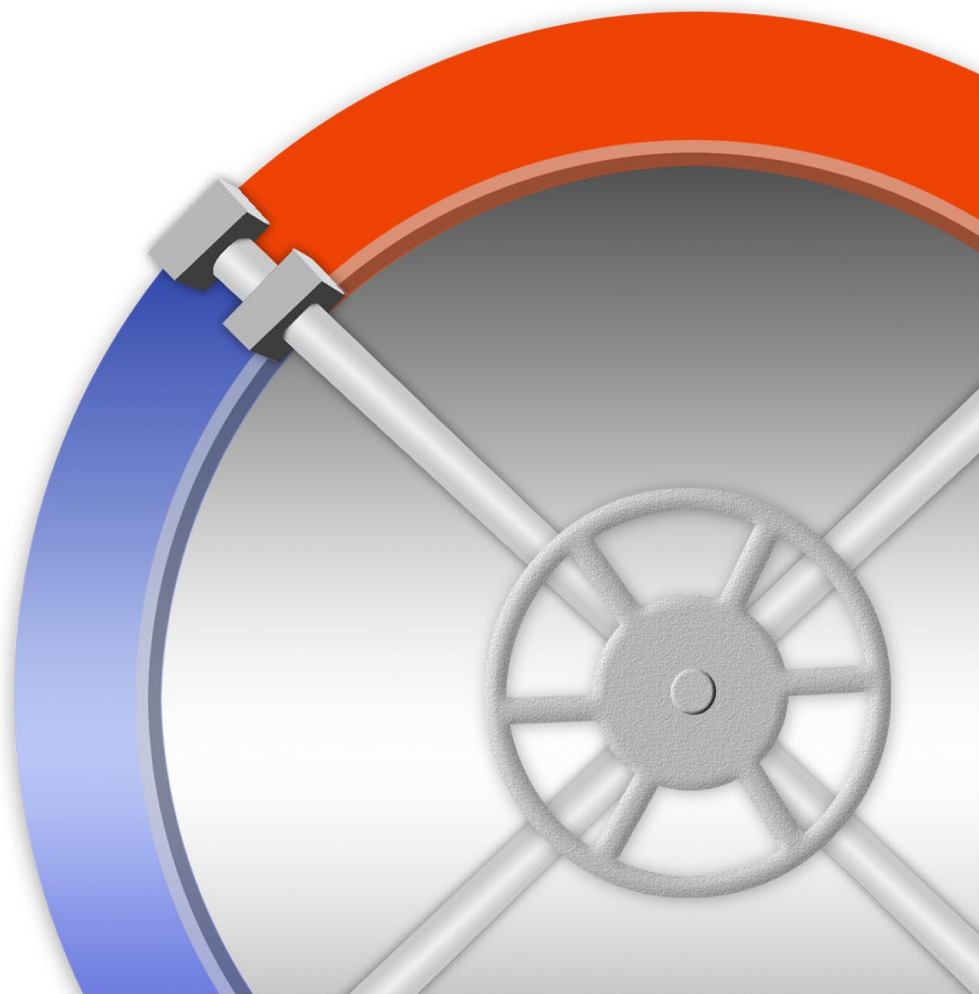




# Users Manual

# Securells™

Application Firewall



## Warranty

This document is supplied on an "as is" basis with no warranty and no support.

## Limitations of Liability

In no event shall eEye Digital Security be liable for errors contained herein or for any direct, indirect, special, incidental or consequential damages (including lost profit or lost data) whether based on warranty, contract, tort, or any other legal theory in connection with the furnishing, performance, or use of this material.

The information contained in this document is subject to change without notice.

No trademark, copyright, or patent licenses are expressly or implicitly granted (herein) with this manual.

## Disclaimer

All brand names and product names used in this document are trademarks, registered trademarks, or trade names of their respective holders. eEye Digital Security is not associated with any other vendors or products mentioned in this document.

## SecureIIS™ Application Firewall Users Manual

© 2002 eEye Digital Security. All rights reserved. | S-M-0702

This document contains information, which is protected by copyright. No part of this document may be photocopied, reproduced, or translated to another language without the prior written consent of eEye Digital Security.

## Collateral Information

For the latest updates to this document, please visit:  
<http://www.eeye.com/partners>

# Table of Contents

<b>Table of Contents</b> .....	<b>ii</b>
<b>Welcome to Security</b> .....	<b>1</b>
<b>System Requirements</b> .....	<b>3</b>
<b>Installation</b> .....	<b>4</b>
Installing SecureIIS 2.0 .....	4
Uninstalling SecureIIS .....	4
<b>Getting Started</b> .....	<b>5</b>
Running SecureIIS for the First Time .....	5
Security Console .....	5
<b>Site Security</b> .....	<b>7</b>
Configuring your Site Security .....	7
Buffers .....	9
Methods .....	12
Keywords .....	15
Shellcode .....	15
Protect .....	16
Web Applications .....	18
Errors .....	18
Folders .....	18
<b>File Monitor</b> .....	<b>19</b>
Using File Monitoring .....	19
<b>Site Statistics</b> .....	<b>22</b>
Using Site Statistics .....	22
<b>Log Viewer</b> .....	<b>23</b>
Using the Log Viewer .....	23
<b>Central Policies</b> .....	<b>24</b>
Using Central Policy Management .....	24
<b>Troubleshooting</b> .....	<b>26</b>
<b>License Management</b> .....	<b>27</b>
<b>Appendix A</b> .....	<b>28</b>
References .....	28

## Welcome to Security

What makes SecureIIS different from other security products?

In June of 1999, a newly founded eEye Digital Security discovered a vulnerability in Microsoft's premier web server, Internet Information Server (IIS). The IIS vulnerability allowed an attacker to remotely execute code on any susceptible IIS web server and gain complete control of the machine. Because a fix for this hole was not immediately available from Microsoft almost 90% of the IIS servers worldwide sat vulnerable and defenseless as the patch was developed.

While patiently waiting out the delay, eEye pondered the idea of a "universal patch" that could reside within the web server application and protect it from all future vulnerabilities of this nature. Out of this thinking came "Ogle", a simple filter created for IIS that limited the length of data that could be passed into the web server, thus preventing a "buffer overflow". Because of this filtering, the tool was able to protect the unpatched vulnerability, as well as any future software flaws that would be susceptible to buffer overflows. This set the stage for the development of a new class of security product – one that intimately understood the application it was protecting, recognized that most attacks against the application could be grouped into similar categories or "classes", and had the ability to prevent these classes of attack from reaching the application. This revolutionary concept led to the development of SecureIIS.

SecureIIS was the first product to introduce the concept of security at the "application layer", which allows the product to work side by side with the application it is protecting, preventing attacks of all types from penetrating unpatched web servers. Building on the original buffer-overflow layer of protection in Ogle, SecureIIS incorporated several more layers of protection that encompassed almost every IIS vulnerability to date. A user-friendly configuration interface and robust logging were added, and in May of 2001 SecureIIS was released.

In July of 2001, eEye again contacted Microsoft regarding a security hole that was even more severe and widespread than the first. Microsoft released a patch in a timely manner, but despite attempts by Microsoft and eEye to notify the public about the potential risks of the vulnerability, over a month later there were over 1 million Microsoft web servers still unpatched and vulnerable.

In the early morning of August the 3rd 2001, eEye received information from a customer that there was a possible Internet worm crawling through his network. After in-depth research on the worm, eEye concluded that the worm was exploiting the vulnerability discovered a month earlier and that it was poised for a massive infestation of the Internet. Despite widespread press coverage, and a growing number of infections, many of the unpatched servers on the Internet remained unpatched. Over the next several months, the worm, dubbed "CodeRed" by eEye, proceeded to pummel corporate networks and caused an estimated 7 billion dollars in damage.

Had any of the unpatched web servers been running SecureIIS they would have been proactively protected from CodeRed and all of its successors (e.g. CodeRed II, Nimda, and more) — long before the worms were even created.

### **Importance of Application Layer Protection**

While Code Red was making headlines, it was clearly evident why SecureIIS was destined to become such a critical piece of web server security. Since web servers need to remain open to the outside world, security methods of blocking and restricting access via traditional firewalls and intrusion detection systems can only be applied selectively and in reaction to an attack. A better, more proactive solution created specifically for IIS was needed, and eEye delivered.

A research and development powerhouse, eEye has a long track record of stretching the limits of security. eEye employs some of the best and brightest security professionals in the world, and it is the company's absolute commitment to security that fuels the continual discovery of software vulnerabilities as well as the development of ground-breaking products to protect and prevent those vulnerabilities.

Please send suggestions, updates, and comments to:

**eEye Digital Security**  
<http://www.eEye.com>  
[info@eEye.com](mailto:info@eEye.com)

## System Requirements

### Software:

- Microsoft Windows NT 4.0
- Service Pack 6a or higher
- Microsoft Internet Information Services 4.0
  
- Microsoft Windows 2000
- Service Pack 2 or higher
- Microsoft Internet Information Services 5.0
  
- Microsoft Windows XP
- Microsoft Internet Information Services 5.1

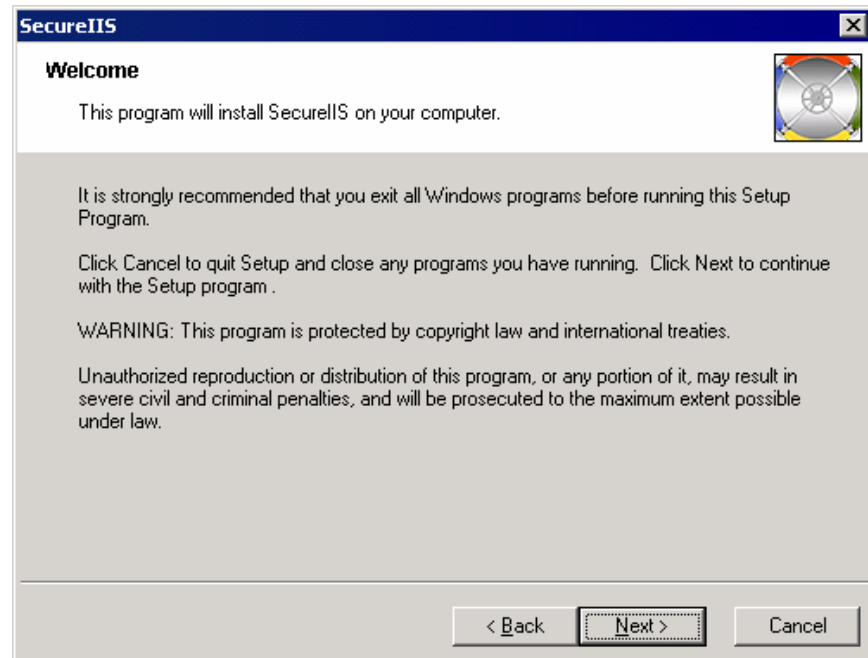
### Hardware:

- 128 MB RAM
  
- 8 MB of Hard Disk Space

# Installation

## Installing SecureIIS 2.0

The installation of SecureIIS is very straightforward. After accepting the license agreement and clicking [NEXT] on the initial screen, you will receive the option to choose an installation directory for SecureIIS. We recommend using "C:\Program Files\eEye Digital Security," which is selected by default. It may be appropriate, however, for you to install SecureIIS to a different location depending on your server configuration.



The SecureIIS 2.0 installer allows you to import the configuration from previous versions of SecureIIS. This will allow you to upgrade SecureIIS while keeping all of your customized settings intact.

After the installation is complete you can begin to use SecureIIS 2.0. The SecureIIS interface can be accessed from:  
Start Menu -> Programs -> eEye Digital Security -> SecureIIS

Please see the next section, Getting Started, for more information on using SecureIIS 2.0 for the first time.

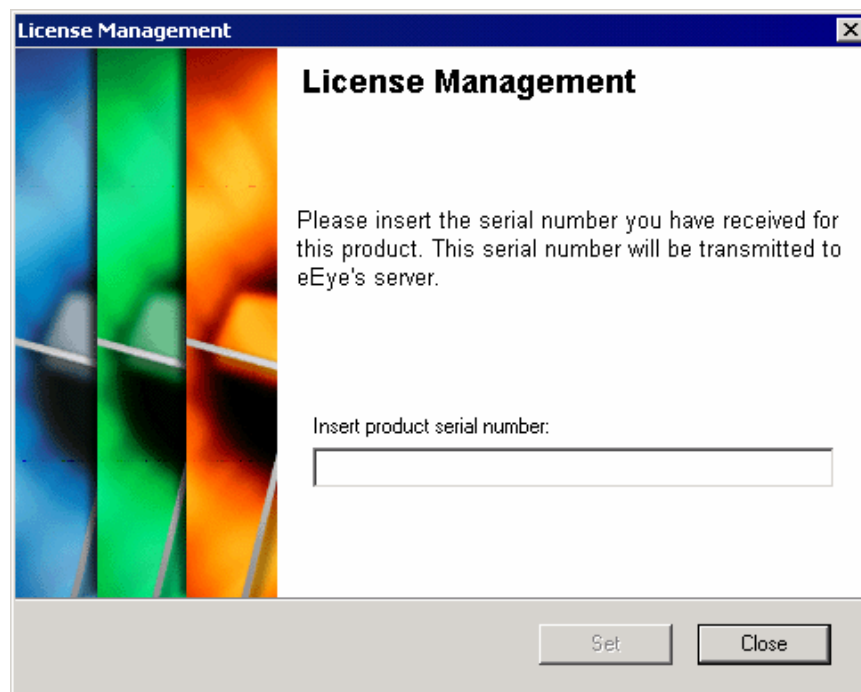
## Uninstalling SecureIIS

To uninstall SecureIIS, click on the Windows "Start" button, go to "Settings", and click on "Control Panel". Select the "Add/Remove Programs" option. Follow the on-screen instructions to remove SecureIIS from your system. You will be required to restart your system to complete the SecureIIS uninstall process.

## Getting Started

### Running SecureIIS for the First Time

The first time you run SecureIIS 2.0 you will be prompted for a serial number. Please enter your serial number and click the set button. After you have entered your serial number the SecureIIS console will load.



If you have any troubles with your serial number or using the product, support can be reached using one of the following methods:

#### Frequently Asked Questions

<http://www.eeye.com/html/Support/FAQ/SecureIIS.html>

#### Support Request Form

<http://www.eeye.com/html/Support/Request/index.html>

### Security Console

SecureIIS and other eEye products run inside of the eEye Security Console. The Security Console allows you to configure and maintain your web server's security quickly and efficiently. The Security Console also allows engineers at eEye to easily incorporate new features into SecureIIS when they are developed. These features are called "components". Each component has a unique functionality that adds to the usability and power of the eEye Digital Security Console and eEye products.

SecureIIS 2.0 includes the following components:

### Site Security Management

SecureIIS has undergone many exciting changes since version 1.0. In version 2, general configuration is much more robust, and users can configure security settings for entire sites down to individual directories. This feature gives users a more granular level of control over their sites' security. The added level of control allows Administrators to configure "focal points" within a website where security needs to be tighter, and also allows the relaxation of security in certain areas where it may not be as crucial.

### File Monitoring

The File Monitoring system provides SecureIIS the ability to monitor file system activity. File or directory creation, deletion or modification can be observed and recorded.

### Site Statistics

Users can now view statistical information about website usage. Valid requests and failed attacks are organized in easy-to-understand graphs to give administrators a general overview of their sites' activity.

### Log Viewer

A new feature in SecureIIS 2.0, the log viewer allows the administrator to view log files from the console. In previous versions of SecureIIS the administrator had to open a text file to view failed requests. The Log Viewer provides advanced sorting and exporting of log entries.

### Site Security

Using the Site Security Component Interface, you can configure security settings for every site and folder on your web server. The Site Security Configuration menu gives you total control over your web server and how it interacts with users. You can do much more than just tighten your web server security policy. In the following section we will go over general usage and also provide some ideas on how you can make SecureIIS work to fit your needs.

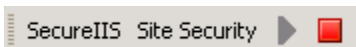
### Enabling and Disabling SecureIIS Protection

To enable and disable SecureIIS filtering and general protection you can use the site security toolbar:

**When SecureIIS is disabled the green arrow will be visible:**



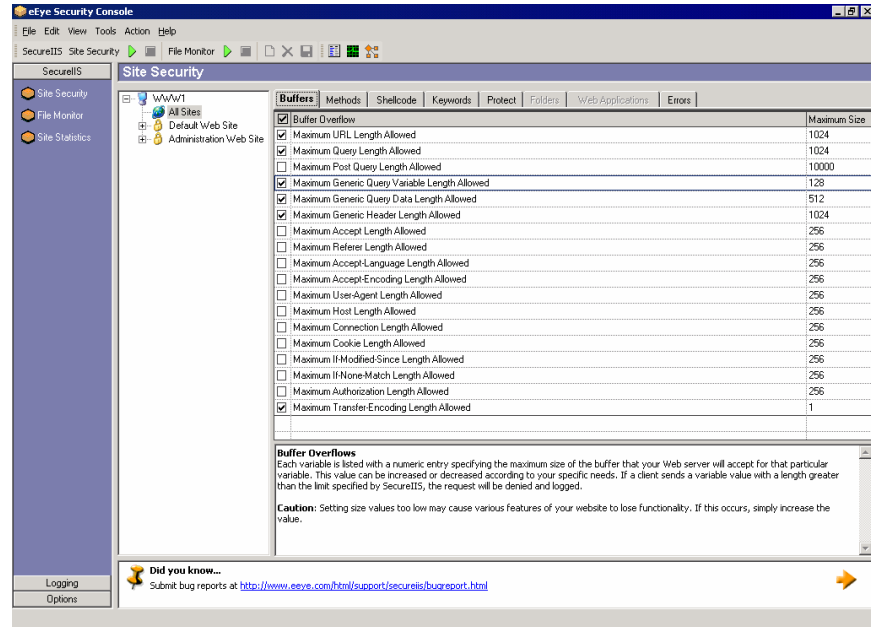
**When enabled the "Stop" button will be visible:**



# Site Security

## Configuring your Site Security

This is the general SecureIIS Site Security interface. From here you can select a website, or a specific folder within a website to configure.



After you have selected a folder or site to configure, you can configure various components by clicking one of the Property Tabs to the right. Each tab has a suite of options available to increase or decrease security on a web server. We recommend leaving the default settings intact for the highest level of security. But if you would like to alleviate some of the security offered by SecureIIS you can do so by disabling features using the property tabs.

We will now give a brief description of the property tabs available in SecureIIS 2.0. More detail on each section follows the overview.

**Note:** When making changes remember to use the Apply button to save your changes and apply them to the active SecureIIS filter.

The following tabs are available in SecureIIS 2.0.



### Buffers

The settings in the Buffers menu allow the administrator to restrict the size of data that is allowed to be passed to the web server from a browser or other HTTP client application.

### Methods

This property tab is used to allow or restrict the use of various HTTP methods such as “GET”, or “POST”. Other methods such as PROPPATCH, LOCK or UNLOCK should only be used with Development and Authoring, or WEBDAV. WEBDAV is primarily used to upload and alter content on the web server from a remote location.

### Shellcode

By enabling these options, SecureIIS will prevent an attacker from using typical buffer overflow exploits against a web server. During the exploitation of a buffer overflow vulnerability, an attacker needs to send a “payload” that consists of shellcode. A payload is basically a small application that is designed to give an attacker additional access to the target when executed.

### Keywords

The Keywords menu prevents various keywords submitted by website visitors from reaching the web server. This method can be used to prevent common SQL injection techniques, buffer overflows, format bug exploitations, and even for content filtering (to limit profanity entered by users). Keep in mind that this feature can be used to block any request that contains a certain keyword.

### Protect

The Protect property tab allows the administrator to quickly disable unused data encoding schemes that can be used to bypass security systems such as the %u encoding IDS bypass vulnerability.

### Folders

This tab can be used to configure access to site folders and directories. For example, if a folder on the website contains data that should not be available to external web clients, an administrator can restrict access to it by un-checking the box next to its entry in the “Allowed Folder” list.

### Web Applications

If a SecureIIS customer wishes to use web server applications such as FrontPage or Outlook Web Access, he or she can use this menu to quickly configure SecureIIS to allow these applications to function correctly. To enable these services click the checkbox next to the application. This will release certain security constraints within SecureIIS and allow the applications to function without SecureIIS interfering.

### Errors

The administrator can use the Errors property tab to configure SecureIIS to return custom error pages for web errors. This option is available for sites that wish to control their general flow of traffic into support or help pages. The administrator can configure SecureIIS to use custom 404 or 406 error pages.

## Buffers

Recommended Setting: Enable All

Almost every web server on the market, at one point or another, has been susceptible to buffer-overflow vulnerabilities. By checking the length of input from the client, SecureIIS can stop old buffer overflow exploits from working and also prevent future exploits.

In order for most buffer overflow exploits to work, an attacker needs to send a piece of executable code (called a “payload”) along with the exploit. If the sizes of incoming HTTP elements are limited, it becomes very difficult for an attacker to find a place to put his payload. Most payloads in Win32 environments are at least 1000 bytes.

In this section of the interface, each variable is listed with a numeric entry specifying the maximum size of the buffer that your web server will accept for that particular variable. This value can be increased or decreased according to your specific needs. If a client sends a variable value with a length greater than the limit specified by SecureIIS, the request will be denied and logged.

Caution: Setting size values too low may cause various features of your website to lose functionality. If this occurs, simply increase the value.

### Maximum URL Length Allowed

Recommended Setting: 1024

This value specifies the maximum possible length of the entire Uniform Resource Locator or URL (see the HTTP protocol documentation included for more information).

### Maximum Query Length Allowed

Recommended Setting: 1024

By lowering this value you can change the maximum query size that SecureIIS allows to be passed to IIS. CGI scripts and other web programs use query strings to get input from a client browser.

```
http://localhost/scripts/secureiis?First=john&Last=doe&Job=vd
```

The query portion of this request is:  
First=john&Last=doe&Job=vd

### Maximum POST Query Length Allowed

Recommended Setting: 10000

This specifies the maximum possible length (in bytes) of the POST query data allowed. Some web appliances use POST queries to upload files; if this is the case for your server, you may want to set this value higher than 10000. SecureIIS will stop anyone from uploading a file (or

submitting a form with the POST method) larger than the amount of bytes specified.

#### **Maximum Generic Query Variable Length Allowed**

Recommended Setting: 128

This setting specifies the maximum length of a variable name being passed via a query. For example: “test” and “test1” are the variable names in the following query:

```
http://localhost/scripts/example.exe?test=data&test1=data
```

#### **Maximum Generic Query Data Length Allowed**

Recommended Setting: 512

This setting specifies the maximum length of the data being supplied for a specific variable. For example “test” and “test1” are variable data entries in the following query:

```
http://localhost/scripts/example.exe?variable1=test&variable2=test1
```

#### **Maximum Generic Header Length Allowed**

Recommended Setting: 1024

The HTTP client request header contains all of the information the server needs to process the request.

#### **Maximum Accept Length Allowed**

Recommended Setting: 256

The “Accept” client-request header field can be used to specify certain media types that are acceptable for use by the client.

#### **Maximum Referer Length Allowed**

Recommended Setting: 256

The “Referer” client-request header field allows the client to specify, for the server’s benefit, the address of the web page that is responsible for directing the client to the current resource. (Yes, “Referer” is misspelled, as specified in RFC 2616)

#### **Maximum Accept-Language Length Allowed**

Recommended Setting: 256

The “Accept-Language” request header specifies the set of natural languages that are preferred by the client making the request.

#### **Maximum Accept-Encoding Length Allowed**

Recommended Setting: 256

The “Accept-Encoding” request header restricts the content-coding types that are acceptable for the client making the request.

#### **Maximum User-Agent Length Allowed**

Recommended Setting: 256

The “User-Agent” request header field contains information about the user agent originating the request, usually browser version or browser type.

#### **Maximum Host Length Allowed**

Recommended Setting: 256

The “Host” request header field specifies the Internet host and port number of the resource being requested by the client.

#### **Maximum Connection Length Allowed**

Recommended Setting: 256

The “Connection” field allows the client or server to specify what is desired for the current connection such as “Keep-Alive”, a mode where the client can send more than one session during the HTTP transaction.

#### **Maximum Cookie Length Allowed**

Recommended Setting: 256

The “Cookie” variable is used by a web server when it needs to keep state information about a current web connection.

#### **Maximum If-Modified-Since Length Allowed**

Recommended Setting: 256

The “If-Modified-Since” method is used as follows: If the resource requested has been modified since the timestamp provided, return the resource to the client. Otherwise return a 304 error (Not Modified).

#### **Maximum If-None-Match Length Allowed**

Recommended Setting: 256

The “If-None-Match” request entity is used as follows: if any of the entity tags provided to an “If-None-Match” request exist, the server will not process the request.

#### **Maximum Authorization Length Allowed**

Recommended Setting: 256

The “Authorization” request entity holds authorization credentials such as a base64-encoded username and password or an NTLM hash.

## Methods

Recommended Setting: GET and POST enabled only.

By allowing only the most common HTTP methods, SecureIIS can prevent many potential security risks involving the handling of unexpected HTTP request methods.

**Note:** FrontPage and other WWW publishing tools need access to non-standard methods. If you experience problems while using these applications, select the “Enable FrontPage Extensions” option in the “Web Applications” section of the SecureIIS interface.

### GET

Recommended Setting: Enabled

The GET command is used to request files from your web server.

### POST

Recommended Setting: Enabled

The POST command is used when users submit information via forms or queries within your website.

### PUT

Recommended Setting: Disabled

The PUT command is used for transferring files to your web server.

### OPTIONS

Recommended Setting: Disabled

The OPTIONS command allows for someone to remotely gain a list of all commands that your web server understands. Typically it's not necessary to have this command enabled.

### HEAD

Recommended Setting: Disabled

The HEAD command allows remote users to check the server version and also gather other information sent back to the client in the response header. Many times attackers use this as a reconnaissance tool to learn what is or is not installed on your website. There typically isn't any need for remote users to have access to the HEAD command.

### DELETE

Recommended Setting: Disabled

The DELETE command allows remote users to attempt to delete files from your web server. We highly recommend not allowing this command to be executed on your server. However, there are some products, such as FrontPage Server Extensions, that require this command to be enabled.

### **COPY**

Recommended Setting: Disabled

The COPY command allows remote users to copy files to your web server. Once again, unless you have a specific need for this command, we suggest turning it off.

### **MOVE**

Recommended Setting: Disabled

The MOVE command allows users to move files around on your web server. We recommend not allowing this command to be enabled.

### **MKCOL**

Recommended Setting: Disabled

The MKCOL command allows files and folders to be created on web servers that support the new Web DAV protocols.

### **PROPFIND**

Recommended Setting: Disabled

The PROPFIND command allows the properties of files and folders to be retrieved from web servers that support the new Web DAV protocols.

### **PROPPATCH**

Recommended Setting: Disabled

The PROPPATCH command allows the properties of files and folders to be changed or removed from web servers that support the new Web DAV protocols.

### **LOCK**

Recommended Setting: Disabled

The LOCK command allows files, folders and resources to be locked on web servers that support the Web DAV protocols.

### **UNLOCK**

Recommended Setting: Disabled

The UNLOCK command allows files, folders and resources to be unlocked on web servers that support Web DAV protocols.

**SEARCH**

Recommended Setting: Disabled

The SEARCH command allows for server-side searches to be performed on web servers that support Web DAV protocols.

**BCOPY**

Recommended Setting: Disabled

The BCOPY command allows remote users to batch copy files to your web server. This method is an undocumented extension by Microsoft to Web DAV.

**BDELETE**

Recommended Setting: Disabled

The BDELETE command allows remote users to batch delete files on your web server. This method is an undocumented extension by Microsoft to Web DAV.

**BMOVE**

Recommended Setting: Disabled

The BMOVE command allows remote users to batch move files on your web server. This method is an undocumented extension by Microsoft to Web DAV.

**BPROPFIND**

Recommended Setting: Disabled

The BPROPFIND command allows the properties of files and folders to be retrieved in batch from web servers that support Web DAV protocols. This method is an undocumented extension by Microsoft to Web DAV.

**POLL**

Recommended Setting: Disabled

The POLL command is used by the HTTPMail protocol extensions. It allows users of HTTPMail to poll a server for new messages.

**SUBSCRIBE**

Recommended Setting: Disabled

The SUBSCRIBE command is used by the Rendezvous Protocol (RVP), an extension to Web DAV. It allows users of instant messaging to begin receiving updates from the server.

**TRACK**

Recommended Setting: Disabled

The TRACK command allows changes to be detected from web servers that support Web DAV protocols. This method is an undocumented extension by Microsoft to Web DAV.

### **UNSUBSCRIBE**

Recommended Setting: Disabled

The UNSUBSCRIBE command is used by the Rendezvous Protocol (RVP), an extension to Web DAV. It allows users of instant messaging to stop receiving updates from the server.

### **CONNECT**

Recommended Setting: Disabled

The CONNECT command is used to make HTTPS (secure HTTP) connections through a proxy server.

## **Keywords**

Recommended Setting: SYSTEM32 and cmd.exe.

By searching for certain keywords, SecureIIS can identify requests that could potentially cause a security risk. If a client makes a request that contains one of the selected keywords, the request will be denied and logged. Many of the tools attackers use try to execute a command shell, which is by default located in C:\WINNT\system32\cmd.exe. By searching for "system32\cmd.exe", "cmd.exe", or "system32", we can greatly reduce the risk of a typical exploitation.

Caution: Be mindful of the information that you choose to filter. The more keywords to be filtered for, the higher the load placed on SecureIIS.

## **Shellcode**

Recommended Settings: Enable all; Disable if web server is non-English.

Caution: Some multilingual sites will not operate correctly if Shellcode Protection is enabled. This is due to the fact that various characters in foreign alphabets are outside the ASCII range, making them high-bit. If you have frequent problems with VerifyHIGHBIT error, and your site is multilingual, you may need to disable these features to regain full functionality of your website.

Caution: If your clients or customers upload binary files to your website, or need to add email attachments to their emails using Outlook Web Access, you may need to disable these features due to the fact that the files being uploaded may contain high-bit data.

### **High-Bit Shellcode Protection in URL**

Recommended Setting: Enabled

This setting enables SecureIIS to search through the effective URI or URL for high-bit data that resembles Shellcode (often used in an attacker's payload). Enabling this feature allows SecureIIS to drop any connection that contains high-bit data in the URI/URL portion of the client request. (Refer to the HTTP Protocol documentation for more information about URI's or URL's)

### **High-Bit Shellcode Protection in Query**

Recommended Setting: Enabled

This setting enables SecureIIS to search through the query string for high-bit data that resembles Shellcode. Enabling this feature allows SecureIIS to drop any connection that contains high-bit data in the query portion of the client request.

### **High-Bit Shellcode Protection in Header**

Recommended Setting: Enabled

This setting enables SecureIIS to search through the effective header for high-bit data that resembles Shellcode. Enabling this feature allows SecureIIS to drop any connection that contains high-bit data in the request header of the client request.

### **High-Bit Shellcode Protection in POST Data**

Recommended Setting: Enabled

This feature searches through the effective POST data of an HTTP request for high-bit data that resembles an attacker payload, or Shellcode. Enabling this feature allows SecureIIS to drop any connection that contains high-bit data in the data portion of a POST method based client request.

## **Protect**

Recommended Setting: Enable All

### **Protect against Directory Traversal Exploits in URL**

Recommended Setting: Enabled

Enabling this option will allow SecureIIS to stop any incoming connection containing data pertaining to directory traversal exploits in the URL portion of the request header. Various server add-on applications such as Cold Fusion have had directory traversal vulnerabilities in the past. Enabling this feature will prevent this and similar holes from being exploited.

### **Protect against Directory Traversal Exploits in Headers**

Recommended Setting: Enabled

Enabling this option will allow SecureIIS to stop any incoming connection containing data pertaining to directory traversal exploits in the request header portion of the request session.

#### Protect against Directory Traversal Exploits in Query String

Recommended Setting: Enabled

Enabling this option will allow SecureIIS to stop any incoming connection containing data pertaining to directory traversal exploits in the query string portion of the request session.

#### Protect against Directory Traversal Exploits in POST Data

Recommended Setting: Enabled

Enabling this option will allow SecureIIS to stop any incoming connection containing data pertaining to directory traversal exploits in the request data portion of the request session.

#### Protect Against Encoding Abuse Exploits In URL

Recommended Setting: Enabled

Enabling this option prevents a malicious user from encoding attacks in the URL portion of their request session. Data encoding is usually an attempt to bypass security measures in place, or to exploit known IIS vulnerabilities dealing with data decoding.

#### Protect Against Encoding Abuse Exploits In Headers

Recommended Setting: Enabled

Enabling this option prevents malicious users from encoding attacks in the request header portion of their request session.

#### Protect Against Encoding Abuse Exploits In Query String

Recommended Setting: Enabled

Enabling this option prevents malicious users from encoding attacks in the query string portion of their request session.

#### Protect Against Encoding Abuse Exploits In POST Data

Recommended Setting: Enabled

Enabling this option prevents malicious users from encoding attacks in the request data portion of their request session.

## Web Applications

Recommended Settings: Disable All

**Note:** Enabling items in the Web Application section will modify other configuration items that are necessary for the application to function.

### Allow FrontPage Server Extensions

Recommended Setting: Disabled

Enabling this option will allow FrontPage server Extensions to be accessed.

Caution: Websites will often be configured in a way that documents will be accessed in a traversal format (relative path) such as “../”, if this is the case, you may need to disable Directory Traversal Exploit protection to regain full functionality.

### Enable Outlook Web Access 2000

Recommended Setting: Disabled

Enabling this option will configure SecureIIS to allow clients to use Outlook Web Access on a web server secured with SecureIIS.

## Errors

The files specified are displayed when SecureIIS detects an attack attempt and blocks access to the requested resource. The 404 error file is displayed instead of the requested URL when a page is not found. The 406 error file is displayed for all other errors.

Double-click an item to change its settings.

## Folders

Recommended Setting: Web root and custom site scripts paths enabled only.

With this feature enabled, SecureIIS can validate that the client is accessing only files that are in a set of allowable directories. The validation occurs before and after IIS processes the HTTP request. This is another great tool for stopping attacks such as “Directory Traversal Attacks”.

We recommend enabling access only for directories that you need; uncheck any directory that your website DOES NOT need to function.

# File Monitor

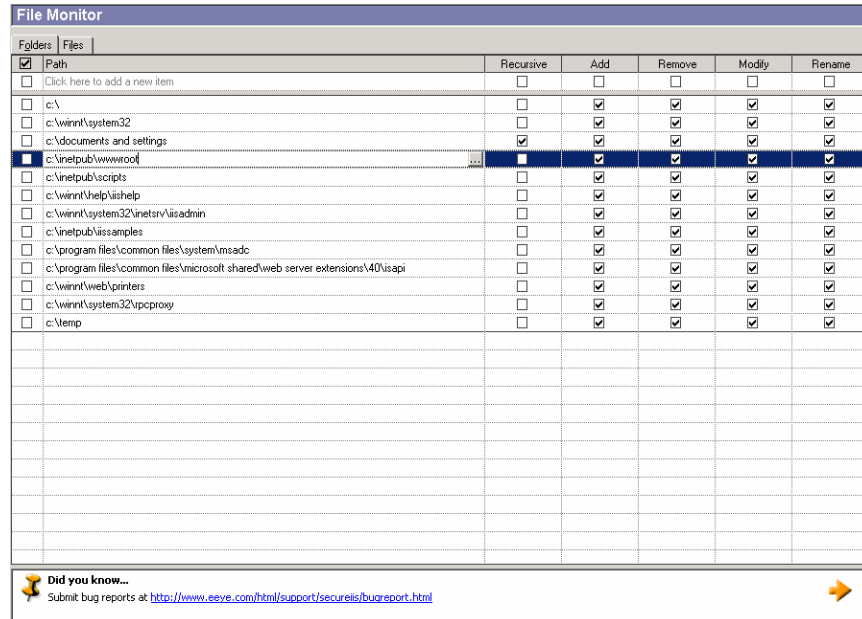
The SecureIIS File Monitoring system allows you to monitor the file system of your web server. File and folder creation, modification and deletion can all be monitored using the File Monitor component. This feature allows you to track possible intrusions and stay alerted to unexpected changes to your website.

Each time the File Monitor notices a change in the monitored file system, a log entry is created that can be viewed using the Log Viewer. Each log entry created by the File Monitor will contain the name of the file or folder that was modified, along with the type of modification that occurred.

See the section of this manual dedicated to the Log Viewer for more information.

## Using File Monitoring

When you open the File Monitor Interface, you will be prompted with the current monitoring configuration. Every entry is preceded by a checkbox. If this checkbox is marked with an X then the file or folder listed is being monitored, otherwise monitoring has been disabled. By default there are several common file and folder entries, but you can easily add your own by following the directions below.



### Enabling the File Monitoring Service

To enable and disable SecureIIS File Monitoring you can use the File Monitor toolbar:

**When File Monitoring is disabled the green arrow will be visible:**



**When File Monitoring is enabled the “Stop” button will be visible:**



### Configuring File Monitoring

To add a new entry, click on the “Click here to add a new item” section of the interface. This section is located at the top of the file and folder list. You can either type in the path to the file or folder you wish to monitor, or click the “...” button to browse for the desired path. After the entry has been added you have the following configuration options:

#### For Folders:

##### Recursion

This option allows you to monitor every file or folder lower in the hierarchy than the current entry. For example, if you added the folder C:\TEST\, then any changes to the folder C:\TEST\TEST2 will also be monitored.

##### Add

By enabling the Add option, SecureIIS will monitor any file or subfolder creations.

##### Remove

By enabling the Remove option, the administrator will be notified of any file or folder deletions.

##### Modify

By enabling the Modify option, SecureIIS will monitor the current folder for any changes or modifications to its files or subfolders.

##### Rename

By enabling the Rename option, the Administrator can be alerted anytime the folder is renamed.

#### For Files:

##### Remove

By enabling the Remove option when enabled will notify the administrator if the file is deleted.

##### Modify

By enabling the Modify option, SecureIIS will monitor the file for any changes or modifications.

**Rename**

By enabling the Rename option, the Administrator can be alerted anytime the file is renamed.

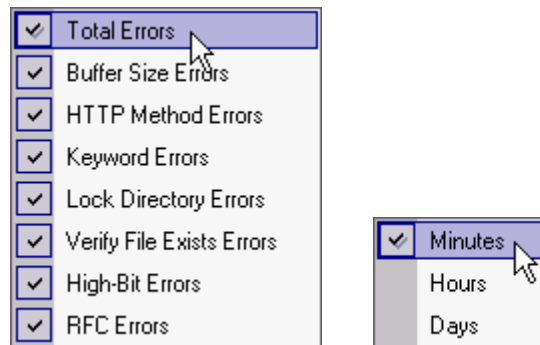
**Caution:** Any time a file is renamed, SecureIIS will no longer monitor changes on that file, so keep in mind that if you decide to change a directory name or filename, you will need to reload those entries into SecureIIS.

# Site Statistics

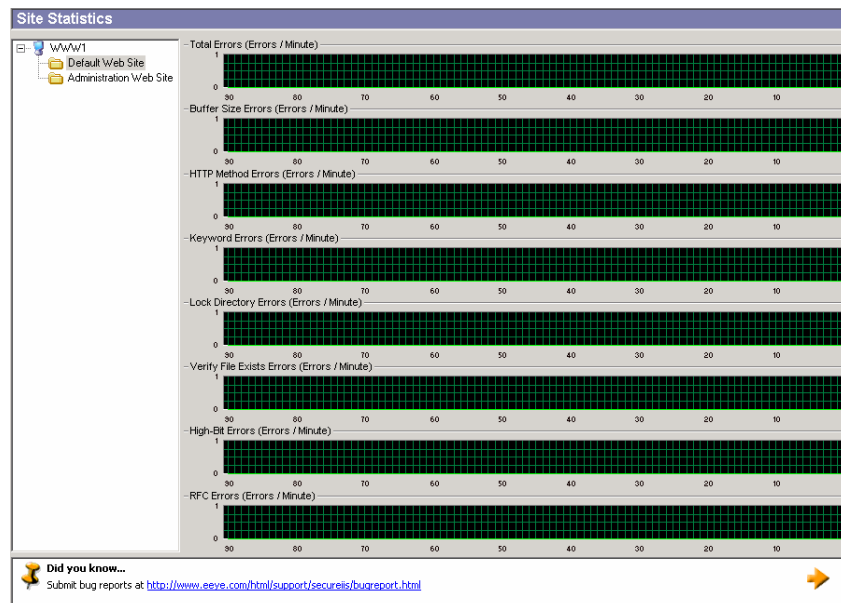
SecureIIS administrators can now view filter statistics using the Site Statistics component in the eEye Digital Security SecureIIS Console. Filtered requests and failed attacks are organized in easy-to-understand graphs that provide a general overview of how SecureIIS is actively filtering incoming traffic.

## Using Site Statistics

Open the View pull down menu in the upper left-hand corner of the eEye Digital Security SecureIIS Console. At the bottom you will see two entries, Site Statistics and Timeframe.

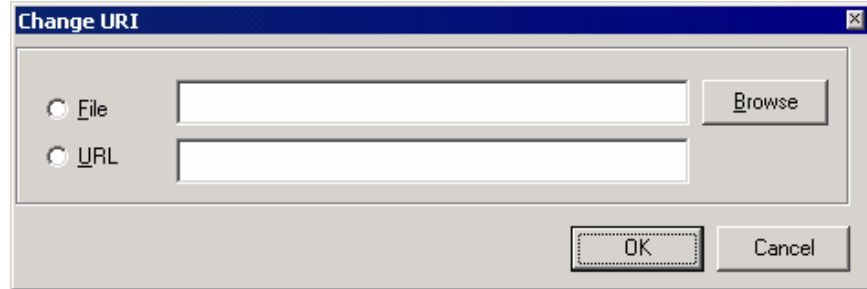


In the Site Statistics menu you can enable and disable the appearance of certain statistics, each entry in this list is a category. By un-checking an entry you can prevent a category of statistics from being displayed in the Site Statistics Component Interface. The Timeframe menu can be used to configure the time range of the report - minutes, hours and days are available options. You can toggle between each of the three settings to find the one that suits your needs the best.









You can use the “Browse” feature to locate an exported policy, or you can enter a URL. The following is an example of policy URL that you can use.

`http://admin.host.com/MasterPolicy.ini`

Click the checkbox in the left column to enable your selected policy. After you have enabled Central Policy support, the master policy file will be imported whenever a change is detected.

**Note:** As each web server contains different website, only global settings from the All Sites section are imported via Central Policies.

## Troubleshooting

In this section we will cover troubleshooting common issues with SecureIIS. New features have been added in SecureIIS 2.0 that make troubleshooting much easier than in previous versions of SecureIIS.

Let use a common example:

You get an email from a customer describing problems accessing a certain part of the website that you are administering with SecureIIS. The customer informs you that he or she is getting a SecureIIS warning when they visit the website. From this point you need to get the error page Reference ID. The Reference ID is a unique identifier that will correlate a log entry with a particular instance of an error. When someone is presented with a SecureIIS error page, a Reference ID is assigned to that particular error and sent back to them in the error page itself. In the error page returned there is a section at the top that will look like the following:

If you feel that you have received this page in error, please contact the administrator of this web site, reporting the following reference ID:

20020510174094

After you have retrieved this reference ID from the customer, you can examine the actual error event using the Log Viewer. When you have found the error in the Log Viewer you can make adjustments as needed by using the Site Security Interface.

If you need additional support, our highly trained technical support staff is available to answer any questions or offer assistance.

### Frequently Asked Questions

<http://www.eeye.com/html/Support/FAQ/SecureIIS.html>

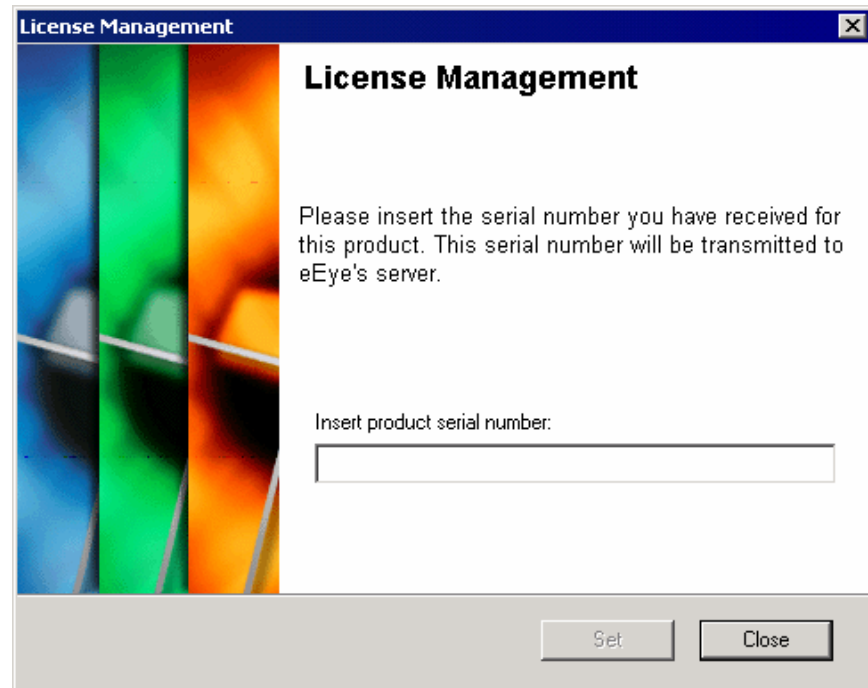
### Support Request Form

<http://www.eeye.com/html/Support/Request/index.html>

# License Management

## Running SecurellS 2.0 for the First Time

When running SecurellS for the first time you will be presented with the serial number registration screen. By entering the serial number you received from eEye Digital Security you will be given access to the SecurellS Console.



After entering in your assigned serial number and clicking the "Set" button, the serial number will be verified with eEye Digital Security and you can begin using SecurellS 2.0.

## Migrating SecurellS to a New Machine

To transfer your SecurellS license to a new machine, launch the License Management interface by selecting "License Management" from the "Help" menu. Next select the "Transfer License" radio button and follow the on-screen instructions.

# Appendix A

## References

### **World Wide Web Consortium - HTTP Specifications and Drafts**

<http://www.w3c.org/Protocols/Specs.html>

### **World Wide Web Consortium - Interesting HTTP Related Papers**

<http://www.w3c.org/Protocols/Papers.html>

### **Hypertext Transfer Protocol -- HTTP/1.1 - Draft Standard RFC 2616**

<http://www.ietf.org/rfc/rfc2616.txt>