

**Invariant Systems  
invURIBL**

**User Guide Version 3.1.1  
10/15/2007**

## Table of Contents

Description:.....	3
New Features In Version 3.0: .....	3
Changes and Bug Fixes in Version 3.1.1:.....	4
Changes and Bug Fixes in Version 3.1.0:.....	4
Changes and Bug Fixes in Version 3.0.7:.....	4
Installation: .....	4
Obtaining a Trial License.....	5
Purchasing invURIBL.....	5
invURIBL Licensing.....	5
Command Line Settings.....	5
Declude Junkmail Integration: .....	6
SmarterMail Integration:.....	6
ORF Integration .....	6
mxGuard .....	6
Configuration Files and Settings:.....	7
invURIBL.exe.Config File Options.....	7
SenderIPWhitelist.txt File Options .....	13
URI-IP-Blackist.txt File Options .....	13
invURIBL-Exceptions.txt File Options .....	14
Performance: .....	14
uriExtract: .....	14
uriTimeAnalyze .....	15
Support:.....	15
Senderbase: .....	16
Program Flow: .....	16
Glossary: .....	16

## Description:

invURIBL is a tool that is used to identify SPAM by extracting URI's (domain names in links) from emails and checking them against URI based blacklists. Our application extends basic URI checking functionality by incorporating features that will allow you to check the URI's IP address and name servers against DNS based blacklists. In addition, we have added a unique feature that allows you to check the URI's IP address and remote mail server against Senderbase, the world's leading email traffic monitoring network.

URI filtering is a very effective anti-spam technique because it focuses on the sole purpose of the spam - the spamvertised link. Today spammers are creating large networks of computers with the sole purpose of distributing SPAM by hijacking virus and trojan infected computers. These networks often grow faster than RBL's are able to list the IP addresses responsible for sending out the SPAM. This is one weakness of traditional RBL's and spammers are actively exploiting this. One advantage of URI filtering over conventional RBL's is that the URI data contained within SPAM tends to be constant where as the source of the message constantly changes.

In addition to checking the URI's against URI blacklists we have added additional features that extend URI filtering. We have incorporated features such as checking the URI's name servers and IP address against blacklists. This is very effective because spammers tend to reuse the same name servers and networks that host their spamvertised sites. We have found that this type of testing is extremely effective at identifying new sources of SPAM.

URI filtering is an essential and effective component to add to your mail server's anti-spam program, because it judges SPAM based on content instead of where it was sent from.

invURIBL will work with any mail server product that can call an external program and process its return code. We have instructions on how to integrate invURIBL into several popular anti-spam products like Declude Junkmail, mxGuard, and Vamsoft's ORF.

## New Features In Version 3.0:

- New URI scanning engine. The new URI scanning engine is faster and more aggressive than the previous version. It now has the ability to detect schemeless domains that were previously undetected (i.e. domains in plain text missing "www" or "http" example: somesite.com).
- Enhanced URI obfuscation decoding. Ability to punish URI's that are obfuscated.
- Completely new mime parser that will handle even the most structurally flawed message.
- Ability to query a URI's name server against multiple RBL's.
- Domains are extracted from email addresses and scanned as well.
- URI IP Blacklist file – If a URI resolves to any of the listed IP blocks weight can be added to the overall message.
- DNS queries are now issued asynchronously.
- URI, Name servers, and RBL return codes can be scored independently.
- Enhanced checking to eliminate double scoring on differing URI's that resolve to the same "A" record or share the same name servers.

- Enhanced name server checking to eliminate double scoring name servers from the same URI on the same RBL list.
- .Net 2.0 Application. invURIBL was rewrote to take advantage of the latest .Net framework features.

### Changes and Bug Fixes in Version 3.1.1:

- Fixed mime corruption issue with accented characters.
- New name server skip option (Bit 4). Allows skipping of the name server check if the URI's name server was listed in any other name server check.
- Updated the TLD list of domains.
- Added an additional URI test (rhsbl.ahbl.org).
- Fixed the "The type initializer for 'x2afa4de559376f87.x223a1e4f63535a54' threw an exception" on x64 bit machines.
- Updated uriExtract and uriTimeAnalyze programs to .Net 2.0.
- Fixed path extraction issue with uriExtract and uriTimeAnalyze.

### Changes and Bug Fixes in Version 3.1.0:

- SmarterMail 4.3.2760+ - invURIBL integrates into SmarterMail natively.
- Custom Headers – invURIBL can now add custom headers to the email message.
- Fixed several mime parsing exceptions for foreign language emails.

### Changes and Bug Fixes in Version 3.0.7:

- Sender IP Whitelist – Fixed a bug where only the first IP address in the SenderIPWhitelist.txt was honored. All of the other entries were not processed.
- Fixed multiple mime parsing exceptions with corrupted messages. If new exceptions are found they will be copied into an /exception folder off the root of where your invuribl.exe file resides.
- When the URI's nameserver tests were disabled they would still be checked.
- Added the following variables that can be used in the log file name. %DD% - Current Day, %MM% - Current Month, and %YYYY% current year.
- Fixed an issue where an exception would occur on a thread performing a DNS lookup and would cause the application to crash to the desktop thus producing a memory dump.

### Installation:

invURIBL can be used as a standalone application or integrated into any mail server package that allows an external application to be called and accepts a return code.

- 1.) Unzip the invURIBL distribution file into the directory where you would like to run invURIBL. Verify that the following files exist.
  - a. invURIBL.exe
  - b. invURIBL-Exceptions.txt
  - c. invURIBL.exe.config

- d. ComponentSpace.Dns.dll
  - e. message.txt
  - f. SenderIPWhitelist.txt
  - g. Uri-ip-blacklist.txt
  - h. uriTimeAnalyze.exe
  - i. uriExtract.exe
- 2.) Obtain a trial license key from <http://www.invariantsystems.com/invURIBL/invURIBLTrial.aspx>
  - 3.) Enter your trial license key into the invURIBL.exe.Config file. This file can be edited with a text file editor like notepad.
  - 4.) At a command prompt in the same directory where you unzipped the invURIBL files run the following command.  
invURIBL.exe 0 0.0.0.0 <Full Path To>message.txt
  - 5.) After you run the above command a log file should be created with the following name URIBL-LogfileMMDD.TXT. Where MM = Current Month and DD = Current Day.
  - 6.) Inside the log file you should see an entry like the following:  
2007-03-26 13:39:16.358 2007-03-26 13:39:16.618 message.txt surbl-org-permanent-test-point.com 127.0.0.126 URI from message body found in multi.surbl.org [126] [Total Weight=34]
  - 7.) If at any step you are prompted with an error please contact support[at]invariantsystems[dot]com for assistance in resolving the issue. If you are not prompted with any errors your installation of invURIBL is working properly.

## Obtaining a Trial License

To obtain a trial license for invURIBL please visit the following link - <http://www.invariantsystems.com/invURIBL/invuribltrial.aspx>. Trial licenses are valid for 30 days.

## Purchasing invURIBL

invURIBL is a very cost effective anti-spam detection program. Single copies are priced at \$50.00 with discounts when you order multiple copies. Email based support and upgrades are provided free of charge for one year from the date of your purchase.

## invURIBL Licensing

invURIBL is licensed using the per server model. For example if you would like to run invURIBL on 5 servers you would need to purchase 5 copies of the program.

## Command Line Settings

To run invURIBL.exe from the command line the syntax is as follows.

invURIBL.exe <CURRENT WEIGHT> <REMOTE MAIL SERVER IP> <PATH TO MESSAGE>

<CURRENT WEIGHT> - The current weight of the message. If you are running invURIBL as a standalone application you can just pass in the value zero.

<REMOTE MAIL SERVER IP> - The IP address of the remote mail server. If you are running invURIBL as a standalone application you can pass in 0.0.0.0.

<PATH TO MESSAGE> - The path and filename of the message to be processed.

Example:

```
invURIBL.exe 0 1.2.3.4 C:\SPOOL\MESSAGE.SMD
```

## **Declude Junkmail Integration:**

invURIBL easily integrates with Declude Junkmail. Once you have completed the steps outlined in the installation section of this manual you will need to integrate invURIBL into Declude.

1.) Edit your Declude global.cfg file and add the following line.

```
INV-URIBL external weight "X:\INVURIBL\INVURIBL.exe %WEIGHT% %REMOTEIP%" 0 0
```

This line configures invURIBL as an external test for Declude that will add the weight generated inside of invURIBL into Declude. **Please make sure you change the path to reflect the path on your system.**

2.) Edit your \$default\$.junkmail files to add the INV-URIBL test.

## **SmarterMail Integration:**

invURIBL easily integrates with SmarterMail version 4.3.2760 and above. Once you have completed the steps outline in the installation section of this manual you will need to integrate invURIBL into SmarterMail.

For SmarterMail integration please see the PDF file invURIBL Installation Instructions For SmarterMail 4.pdf.

## **ORF Integration**

invURIBL easily integrates with Vamsoft's ORF Enterprise Edition 2.1. Once you have completed the steps outlined in the installation section of this manual you will need to integrate invURIBL into ORF.

In the contents of the latest version of invURIBL there will be a folder named ORF. Underneath that folder will contain the following files Readme.txt which is basic installation instructions and a PDF called invURIBL-ORF-Installation.pdf.

Please follow the instructions in those files for integration. If you have any questions at all please contact us and we will help you out.

## **mxGuard**

invURIBL easily integrates with mxGuard 2.0+. Once you have completed the steps outlined in the installation section of this manual you will need to integrate invURIBL into mxGuard.

To enable invURIBL in mxGuard you will need to make two changes to your mxGuard.ini file.

```
[Global]
SpamFilterTypes=Native
to
SpamFilterTypes=Native, invURIBL
```

```
[invURIBL]
PathToEXE=<invuribl program here>
to
PathToEXE=<path on your server>invuribl.exe
```

## Configuration Files and Settings:

invURIBL uses the following configuration files

- **invURIBL.exe.Config** – This is the main configuration file. It contains all of the defined tests and weights that invURIBL will run.
- **SenderIPWhitelist.txt** – This file contains a list of remote mail servers IP addresses that you do not want invURIBL to scan their messages. Individual IP addresses can be listed as well as listing network blocks using proper CIDR notation.
- **invURIBL-Exceptions.txt** – This file contains a list of URI's that you do not want to invURIBL to check. URI's listed in this file will be skipped. Also, URI's that are skipped are not counted towards the max URI's to lookup setting. This helps stops spammers who stuff SPAM with a lot of legitimate domains.
- **URI-IP-Blacklist.txt** – This file contains a list of IP address or IP Subnets that a URI's "A" record is checked against. If a URI's "A" record matches any of the configured addresses than the weight specified in the URI\_BLACKLIST\_WEIGHT key in the invuribl.exe.config file will be added to the messages overall weight.

## invURIBL.exe.Config File Options

invURIBL.exe.Config is the main configuration file. It is an XML based configuration file that can be easily edited with a text editor.

### **Key Format:**

```
<add key="OPTION" value="VALUE" />
```

### **OPTIONS**

- **License\_Key** – This is the license key required to run invURIBL. Trial license keys can be obtained from <http://www.invariantsystems.com/invuribl/invuribltrial.aspx>.  
Example:  

```
<add key="License_Key" value="0000000000000000" />
```
- **Enable Exceptions File** – Enables the use of the URI exception file (invURIBL-exceptions.txt) that contains domains that invURIBL will not test. We recommend that you leave this setting configured as "true".

Example:

```
<add key="Enable Exceptions File" value="true" />
```

- **LogFile\_Path** – Path and filename of the log file that invURIBL will log to. If this key's value is left blank invURIBL will log to the file URIBL-logfile####.txt in the same directory as the invURIBL executable. Please note that if you use #### in the filename it will be substituted with MMDD (Month and Day). The following variables are also available now: %DD% - Current Day, %MM% - Current Month, and %YYYY% Current Year.

Example:

```
<add key="LogFile_Path" value="uribl-logfile%YYYY%%MM%%DD%.txt" />
```

OR

```
<add key="LogFile_Path" value="c:\temp\uribl-logfile####.txt" />
```

- **Log\_Mode** – The logging level that invURIBL will use. Valid options are NORMAL, HIGH, VERBOSE, DEBUG and NONE. \*\*\*NOTE: Debug logging will adversely affect invURIBL performance. We suggest that this log level not be selected unless we request doing so in troubleshooting any issues you may encounter.

Example:

```
<add key="Log_Mode" value="verbose" />
```

- **Skip\_min\_weight** – If the passed in weight is less than this value, invURIBL will exit without running any of the configured tests and return 0 as the exit value.

Example:

```
<add key="SKIP_MAX_WEIGHT" value="-500" />
```

- **Skip\_max\_weight** – If the passed in weight exceeds this value, invURIBL will exit without running any of the configured tests and return 0 as the exit value.

Example:

```
<add key="SKIP_MAX_WEIGHT" value="500" />
```

- **Enable\_Max\_Weight** – If the value is set to true invURIBL will never return a weight higher than the defined MAXWEIGHT key. Values for this key are "true" or "false".

Example:

```
<add key="Enable_Max_Weight" value="true" />
```

- **MAXWEIGHT** – The maximum weight that invURIBL can accumulate throughout running its tests. If the MAXWEIGHT value is exceeded and "Enable\_Max\_Weight" is set to "true" invURIBL will return the MAXWEIGHT value.

Example:

```
<add key="MAXWEIGHT" value="10" />
```

- **MINWEIGHT** – If the accumulated is greater than zero and less than the value set for MINWEIGHT the MINWEIGHT value will be returned. A value of zero disables this feature.

**This feature is new in version 2.5.**

Example:

```
<add key="MINWEIGHT" value="0" />
```

- **Stop\_At\_First\_Match** - invURIBL will exit when the first domain is listed in any of the configured URI or RBL lists. Values for this key are "true" or "false".

Example:

```
<add key="Stop_At_First_Match" value="false" />
```

- **Extract\_Email\_Address\_Domains** - invURIBL will scan the message body looking for email addresses. If an email address is found the domain will be extracted and checked as if it was found in a URI. Values for this key are "true" or "false". **This feature is new in version 3.0.**

Example:

```
<add key="Extract_Email_Address_Domains" value="true" />
```

- **Max\_URI\_Links** – The maximum number of URI links that will be checked. When this value is reached invURIBL will stop checking the remaining links and return the current weight. Any domain that is listed in the exception file will not toward the max URI link value. **This feature is new in version 2.5.**

Example:

```
<add key="Max_URI_Links" value="20" />
```

- **DNS\_Server** – The DNS server that invURIBL will use. If left blank the default Windows DNS Server will be used, but an error will be logged to the log file indicating this field is blank. **This feature is new in version 2.6**

Example:

```
<add key="DNS_Server" value="xxx.xxx.xxx.xxx" />
```

- **DNS\_Server\_Timeout** – The number of seconds that invURIBL will wait for a response from the DNS server.

Example:

```
<add key="DNS_Server_Timeout" value="1" />
```

- **Max\_Message\_Size** – If the message to be processed by invURIBL exceeds the size in KB's specified by this key invURIBL will not process this message. A value of zero will disable this feature and all messages regardless of size will be processed by invURIBL. The value for this key is specified in kilobytes ( 1000 = 1MB).

Example:

```
<add key="Max_Message_Size" value="300" />
```

- **Program\_Timeout** – If invURIBL exceeds the amount of time (in seconds) configured in this key while processing a message it will exit and return the weight up to where it has processed. The value for this key is seconds.

Example:

```
<add key="Program_Timeout" value="25" />
```

- **Add\_Custom\_HeaderX** – Add Custom Headers allow you to insert customized headers into the message file that is being scanned. This is very useful for anti-spam programs that do not evaluate return codes for external processes. Using this feature you can insert headers into the message that you can filter on. The "X" is an incrementing integer value starting at 1 for each header that you would like to add. So the first header you would like to add would be "Add\_Custom\_Header1" and the second header will be "Add\_Custom\_Header2" etc. For the value portion of the header you can utilize several variables.

- %DATE% - The current date and time.
- %VERSION% - The current version of invURIBL
- %WEIGHT% - The invURIBL weight of the message.
- %RANGE% - The invURIBL spam probability range (LOW, MEDIUM, HIGH). The weights that map to LOW, MEDIUM, and HIGH can be customized through the Custom\_Header\_Spam\_Range\_Low\_Med\_High key.

Example:

```
<add key="Add_Custom_Header_1" value="X-invURIBL-Scan: Scanned by invURIBL %VERSION% on %DATE%" />
```

```
<add key="Add_Custom_Header_2" value="X-invURIBL-Weight: %WEIGHT%" />
```

```
<add key="Add_Custom_Header_3" value="X-invURIBL-Range: %RANGE%" />
```

- **Custom\_Header\_Spam\_Range\_Low\_Med\_High** - If the %RANGE% Variable is used in a custom header this key will be used to map the weights that correspond to LOW, MEDIUM, HIGH. The Value of this key is "LOW,MEDIUM,HIGH" where the range starts at the value entered and ends at the next value. For example the value is shipped at default as "1,5,10". This means that Low = 1 < 5, Medium = 5 < 10, HIGH = 10+. If the final weight of the message is 6 than the %RANGE% value will be MEDIUM.  
Example:  
<add key="Custom\_Header\_Spam\_Range\_Low\_Med\_High" value="1,5,10"/>
- **URI\_BlackList\_Weight** – If a URI’s “A” record matches an entry in the URI-IP-Blacklist.txt file the weight specified in the value of this key will be added to the overall weight of the message.  
Example:  
<add key="URI\_Blacklist\_Weight" value="10" />
- **URI\_Encoded\_Weight** – If a URI has been obfuscated to avoid detection the weight configured in this key’s value will be added to the overall weight of the message. Only the hostname is checked for encoding and not the parameters of the URL.  
Example:  
<add key="URI\_Encoded\_Weight" value="5" />
- **URIBL\_ListX** – This is the URI blacklist that the URI will be checked against. Multiple URI blacklists can be defined by incrementing the “X” value in the key. If you are using multiple URI blacklists the first list must be configured as “1” with the rest of the lists following sequentially.  
Example:  
<add key="URIBL\_List1" value="multi.surbl.org" />
- **URIBL\_Weight\_ListX** – The weight that will be assigned if the URI is listed in the defined URIBL\_ListX.  
Example:  
<add key="URIBL\_Weight\_List1" value="3" />
- **Enable\_Custom\_Bitmask\_Values\_URIBL\_ListX** – Enables you to process the return value of the URIBL\_ListX as a bitmask value. URI blacklists like the SURBL multi list “multi.surbl.org” return bitmasked return codes.  
Example:  
<add key="Enable\_Custom\_Bitmask\_Values\_URIBL\_List1" value="true" />
- **URI\_Bitmask\_BitValue\_Y\_Weight\_URIBL\_ListX** – If you configure a URI blacklist like “multi.surbl.org” which returns a bitmasked return code you may want to weight the individual bits independently. If you are using “multi.surbl.org” please see <http://www.surbl.org/lists.html#multi> for which URI Blacklists correspond to which bitmask values. In the key the “Y” value represents the base2 bit values (1,2,4,8,16,32,64,128) and the “X” value represents the **URIBL\_ListX**.  
Example:  
<add key="URI\_Bitmask\_BitValue\_1\_Weight\_URIBL\_List1" value="0" />  
<add key="URI\_Bitmask\_BitValue\_2\_Weight\_URIBL\_List1" value="7" />  
<add key="URI\_Bitmask\_BitValue\_4\_Weight\_URIBL\_List1" value="2" />  
<add key="URI\_Bitmask\_BitValue\_8\_Weight\_URIBL\_List1" value="5" />  
<add key="URI\_Bitmask\_BitValue\_16\_Weight\_URIBL\_List1" value="3" />  
<add key="URI\_Bitmask\_BitValue\_32\_Weight\_URIBL\_List1" value="7" />

<add key="URI\_Bitmask\_BitValue\_64\_Weight\_URI\_BL\_List1" value="5" />  
 <add key="URI\_Bitmask\_BitValue\_128\_Weight\_URI\_BL\_List1" value="0" />

- **Enable\_URI\_Name\_Server\_Check** – Enables the checking of the URI’s name servers against the configured “Name\_Server\_RBL”. Values for this key are “true” or “false”.

Example:

<add key="Enable\_URI\_Name\_Server\_Check" value="true" />

- **Max\_Name\_Servers\_To\_Check** – The maximum number of name servers that will be checked. ***This feature is new in version 2.5.***

Example:

<add key="Max\_Name\_Servers\_To\_Check" value="3" />

- **Name\_Server\_RBLX** – This is the RBL that the URI’s name servers will be checked against. Multiple name server blacklists can be defined by incrementing the “X” value in the key

Example:

<add key="Name\_Server\_RBL1" value="zen.spamhaus.org" />

- **Bitmask\_Skip\_Options\_Name\_Server\_RBLX** – If this value is set to “true” and the URI is listed in one of the URI blacklists the name server checks will be skipped. Values for this key are the bitmask skip value.

Bitmask Value	Description
0	The name server test will not be skipped.
1	The name server test will be skipped if the URI was listed in any of the defined URI list tests.
2	If a URI has multiple name servers and one of the URI’s were listed in the name server RBL it will skip checking the remaining URI’s name servers against the RBL list. This prevents double scoring.
4	If the URI’s name server was listed in a previous name server check the URI will not be scored against on other name server tests. This prevents double scoring.
8	Reserved for future use.
16	Reserved for future use.
32	Reserved for future use.
64	Reserved for future use.
128	Reserved for future use.

Example:

<add key="Bitmask\_Skip\_Options\_Name\_Server\_RBL1" value="1" />

- **Name\_Server\_Return\_Code\_RBLx** – The return code to match from the RBL. Values for this key are the appropriate return code or “\*” for any return code.

Example:

<add key="Name\_Server\_Return\_Code\_RBLX" value="\*" />

- **Name\_Server\_Weight\_RBLx** – The weight that will be added if the name server is listed in the name server blacklist.

Example:

<add key="Name\_Server\_Weight\_RBLX" value="3" />

- **ENABLE\_URI\_IP\_LOOKUPS\_IN\_RBLS** – If the value is set to “true” the URI will be resolved to its IP address and the IP will be checked against any of the defined RBLX lists.

Example:

<add key="ENABLE\_URI\_IP\_LOOKUPS\_IN\_RBLs" value="true" />

- **RBLx** – Specifies the RBL that the URI’s IP address will be looked up in. Multiple RBL’s can be defined. The “x” value in this key is the RBL number. The very first RBL defined must be defined as “1” and the next lists must follow sequentially.

Example:

<add key="RBL1" value="sbl.spamhaus.org" />

- **Return\_Code\_RBLx** – The return code to match from the RBL. Values for this key are the appropriate return code or “\*” for any return code.

Example:

<add key="Return\_Code\_RBLX" value="\*" />

- **WEIGHT\_RBLx** – The weight that will be added if the URI’s IP address is listed in the corresponding RBLx.

Example:

<add key="WEIGHT\_RBL1" value="5" />

- **Bitmask\_Skip\_Options\_RBLx** – This is a bitmask value that gives you the flexibility to determine if you want to run this particular RBLx test should be ran. If you would like to enable multiple skip actions you would add the individual bitmask values. For example if you would like to skip the RBL if either the URI was listed in the URI list or the URI’s name server was listed you would set this option with a value of “3”.

Bitmask Value	Description
0	The RBL test will not be skipped.
1	The RBL test will be skipped if the URI was listed in any of the defined URI list tests.
2	The RBL test will be skipped if the URI’s name server was listed in the defined name server RBL check.
4	Reserved for future use.
8	Reserved for future use.
16	Reserved for future use.
32	Reserved for future use.
64	Reserved for future use.
128	Reserved for future use.

Example:

<add key="Bitmask\_Skip\_Options\_RBL1" value="2" />

- **Enable\_URI\_Senderbase\_Magnitude\_Check** – Enables checking the URI’s IP address against the Senderbase network. Values for this key are “true” or “false”

Example:

<add key="Enable\_URI\_Senderbase\_Magnitude\_Check" value="true" />

- **URI\_Senderbase\_Magnitude\_Threshold** – This is the magnitude threshold that if exceeded the URI\_Senderbase\_Magnitude\_Weight will be added. Please see the section on Senderbase for more details on Senderbase and magnitudes.

Example:

<add key="URI\_Senderbase\_Magnitude\_Threshold" value="50" />

- **URI\_Senderbase\_Magnitude\_Weight** – The weight that will be added if the URI’s magnitude exceeds the URI\_Senderbase\_Magnitude\_Threshold.

Example:

```
<add key="URI_Senderbase_Magnitude_Weight" value="2" />
```

- **Enable\_RemoteMailServer\_Senderbase\_Magnitude\_Check** – Enables the checking of the remote mail server against the Senderbase network. Values for this key are “true” or “false”

Example:

```
<add key="Enable_RemoteMailServer_Senderbase_Magnitude_Check" value="true" />
```

- **RemoteMailServer\_Senderbase\_Magnitude\_Threshold** – This is the magnitude threshold that if exceed the RemoteMailServer\_Senderbase\_Magnitude\_Weight will be added. Please see the section on Senderbase for more details on Senderbase and magnitudes.

Example:

```
<add key="RemoteMailServer_Senderbase_Magnitude_Threshold" value="50" />
```

- **RemoteMailServer\_Senderbase\_Magnitude\_Weight** – The weight that will be added if the remote mail servers magnitude exceeds the RemoteMailServer\_Senderbase\_Magnitude\_Threshold.

Example:

```
<add key="RemoteMailServer_Senderbase_Magnitude_Weight" value="0" />
```

## SenderIPWhitelist.txt File Options

The SenderIPWhitelist.txt file is used to list IP address that you would like invURIBL not to process. Acceptable values in this file can also contain network ranges using proper CIDR notation. Entries must be listed one per line. Comments can also be used throughout the file as long as they are prefaced with the “#” character.

If the remote mail server matches any entries in the file invURIBL will not scan the message and return the value 0.

Example:

```
192.168.125.1
192.168.125.0/24      #Comment
#This is just a line with a comment.
```

## URI-IP-Blackist.txt File Options

If a URI’s “A” record matches an entry in the URI-IP-Blacklist.txt file the weight specified in the value of the “URI\_Blacklist\_weight” key will be added to the overall weight of the message.

Acceptable values in this file can also contain network ranges using proper CIDR notation. Entries must be listed one per line. Comments can also be used throughout the file as long as they are prefaced with the “#” character.

Example:

```
192.168.125.1
192.168.125.0/24      #Comment
#This is just a line with a comment.
```

## invURIBL-Exceptions.txt File Options

invURIBL-Exceptions.txt file is used to list URI's that you would like invURIBL not to process. IP addresses can also be added using reverse notation. For example if you wanted to add the following IP address 1.2.3.4 you would add it in the exceptions file as 4.3.2.1. Comments can also be used throughout this file as long as they are prefaced with the “#” character and are on their own individual line.

### Example:

```
w3.org  
yahoo.com  
aol.com  
hotmail.com  
google.com  
ebay.com
```

## Performance:

The nature of URI filtering is expensive in terms of performance. This is because in order to properly extract URI's the message must be decoded from its current format and the body text must be scanned to extract the URI's. We understand that this process is expensive and we have coded in many options that you can take advantage of to conserve processing resources.

We recommend taking advantage of options like “SKIP\_MIN\_WEIGHT” and “SKIP\_MAX\_WEIGHT”. invURIBL will only process messages if the passed in weight is between SKIP\_MIN\_WEIGHT and SKIP\_MAX\_WEIGHT. Also, setting a MAXWEIGHT for invURIBL is advised as once this WEIGHT is reached the program will exit and stop running additional tests.

You can also specify IP address / Networks of know good server in the SenderIPWhitelist file. If the remote mail server matches any of those entries the message will not be scanned.

Other options we recommend that you enable is skipping name server checking when the URI is listed in the URI list, limit the amount of URI's checked, and maximum name servers to check.

## uriExtract:

uriExtract is a tool that can be used to analyze your invURIBL log files to list the most frequently looked up domains that were not listed in any of the URI lists. This list can then be used to pick out domains that should be added to your invURIBL-Exceptions.txt file. Every domain that uriExtract displays does not mean that it is a legitimate non spam domain it just means that it was not listed in the URI lists.

## **Requirements:**

- uriExtract requires that your logging level be set to verbose for this to work.
- uriExtract must be ran from the same directory where invURIBL was installed on.

**Command Line:**

uriExtract.exe <number of days to process> [limit output to x domains]  
<number of days to process> - Required parameter. This specifies how many previous days to process the logs for. A value of zero will only process today's logs.  
[limit output to x domains] – Optional parameter. This will limit the output of uriExtract to the top x domains.

**Output:**

- uriExtract will output a list of domains and the number of times they were not found in your URI lists to the console.
- uriExtract will also create a text file called uriExtractOutputMMDDYYYY.txt This file will contain just the list of domains that were output to the console. This will hopefully make it easier to add the domains to your invURIBL-Exceptions.txt file.

## uriTimeAnalyze

uriTimeAnalyze is a utility to determine how long invURIBL takes to when processing messages.

**Requirements:**

- uriTimeAnalyze requires that you are running invURIBL log level verbose.
- uriTimeAnalyze must be ran from the directory when invURIBL is installed in.

**Command Line:**

uriTimeAnalyze <number of days of logs to process>  
<number of days of logs to process> - Required parameter. A value of zero will only process the current days logs.

**Output:**

```
X:\IMail\declude\invURIBL>uritimeanalyze 0
Attempting to open: uribl-logfile0815.txt
invURIBL Analysis
-----
Total Messages: 115375
Average Process Time: 00:00:00.3689988
Shortest Process Time: 00:00:00.1870000
Longest Process Time: 00:00:14.1250000
```

## Support:

If you need assistance on invURIBL please contact us at our support email address (support[at]invariantsystems[dot]com) or join our interactive listserv invURIBL invURIBL-list-subscribe[at]invariantsystems[dot]com.

If you need assistance in configuring this application please do not hesitate to let us know.

## Senderbase:

SenderBase is the world's leading email traffic monitoring network, designed to help email administrators research senders, identify legitimate sources of email and stop threats such as spam and viruses. Senderbase's network consists of data provided from over 50,000 organizations that receive email.

We have included a feature into invURIBL that will allow you to query the magnitude of the IP Address of the URI or the remote mail server against the Senderbase network. SenderBase's magnitude is a measure of message volume calculated using a log scale with a base of 10. The maximum theoretical value of the scale is set to 10, which equates to 100% of the world's email message volume (approximately 10 billion messages/day).

We query the Senderbase network extracting daily and monthly magnitude. We then derive a percentage of change based off the IP addresses daily to the monthly magnitude. What we have found through testing is that an IP address that has a high percentage of change in their daily magnitude when compared to their monthly magnitude is a good indicator that the IP address is being used to send SPAM.

## Program Flow:

The following information represents the flow of invURIBL.

- Check Skip Weight
- Check Sender Whitelist File
- Load / MIME decode message
- Extract URI's from message
- Extract domains from Email Addresses
- Check URI against URI List
- Check URI name sever
- Resolve URI to IP address
- Check URI "A" Record against Senderbase
- Check remote mail server against Senderbase
- Return weight

\*\* At each stage MAXWEIGHT and Program Run Time is checked.

## Glossary:

**URI** - Short for *Uniform Resource Identifier*, the generic term for all types of [names](#) and [addresses](#) that refer to [objects](#) on the [World Wide Web](#). A [URL](#) is one kind of URI.

**RBL** - Short for *Realtime Blackhole List*, a list of [IP addresses](#) whose owners refuse to stop the proliferation of [spam](#). The RBL usually lists [server](#) IP addresses from [ISPs](#) whose customers are responsible for the spam and from ISPs whose servers are hijacked for spam relay.

**URIBL** – URI based RBL. It's a realtime blackhole list that contains URI's.

