



MAILWATCHER Filter Engine

MODULE : MWADVT



Lotus Notes Version : 5.x, 6.x, 7.x, 8.x
Engine Version : 4.1.1.126+



Introduction

The MailWatcher Filter Engine, also known as MWADVT (MailWatcher Advanced Task) is able to process email traffic on real time as it goes through router mail.box

This whole system has been designed in order to let you implement your own email policies.

It is a rules interpreter based system.

A rule is a context and a series of actions. Rules have priorities and more than one rule can be applied to a single message.

Email Server can be very stressed especially regarding I/O. The filter engine has its own cache that is maintained and periodically updated.

Some actions are critical, for example, on the fly email copy are secured with readers field, journaling functions are resilient.

The filter engine is one of the MailWatcher modules. It is actually the core one, that makes decisions based on other modules that can process email message before or after it.

Modules that act before are doing content analysis or anti virus scanning, they set verdicts and give informations that become criteria for filtering rules.

Les modules qui se trouvent en aval réalisent des actions relativement complexes, telles que la compression ou des opérations de nettoyage antivirales.

Modules that act after are processing email to do some relative complex tasks like compression or anti-virus clean up actions.

At last, if you configure it, every actions applied to email can be logged into statistics databases.



Rules Context

Before you define the actions of the rules, you need to describe the context, in other words the Sender and/or the Recipients for whom the rules applied.

An email message can be defined like :

- 1 : Sender (From)
- 1,N : Recipient(s) (To)
- N : Attributes, Fields TEXT, TEXTLIST, DATE, NUMBER,...
- 0,N : Attachment(s)

Context is first of all a combination of FROM and TO

MailWatcher analyse each combination as a series of couple

- (Sender * Recipients [i=1,N])

An email sent to 10 people will be process as 10 different messages

For exemple, the couple (FROM*TO(5)) may activate a rule that will not allow the routing to TO(5) but will not prevent the other 9 recipients of receiving the email.

To the context {FROM and TO} you can add other conditions before the rules applied. In fact the real context is a set of 3 boolean functions that must return TRUE to really apply the actions to the message.

Real context = { FROM and TO(i) and Formula }

Formula = set of functions based on Lotus Macro Formula.

if no formula is specified, the code will use @TRUE, that always returns TRUE.



Addresses Parameters and Types

The context is a set of { FROM and TO and Formula }. Now From and To may be of different type. They can be declared as {User, Group, Meta, Meta Group }.

Either the address is a user, or it is a member of a group, or even the address matches a wild-card pattern, or is a member of a group that have entries set as wild-card.

From and To must match the parameters, the type argument is there to define how they have to coincide. In the following, what goes for FROM is true for all the other : TO and Except_From, Except_To.

Type 'User' : the address of the sender - FROM – must be one the alias of the user or the Internet Address or the Forward Email Address

Type 'Group' : the address of the sender must be a member of the declared group. To this group are virtually added all the possible alias, so that the group contains a list of type 'User'

Type 'Meta' : the address of the sender must match the supplied pattern

Type 'Meta Group' : The address of the sender must be/match a member of the specified group in which the member can be defined as wild-card pattern

Addresses Exceptions

The exceptions are evaluated if { FROM and TO and Formula } conditions are TRUE.

Exception can be defined for FROM and/or TO. Respectively Except_From, Except_To

The rules is not applied if one the of exception is TRUE.

That is to say : if {Except_FROM **or** Except_TO} => TRUE



Rules Actions

Actions can be combined. Some of the actions can modify the email, Still the filter engine always maintain a safe copy, for other rules to be applied or not. When no rules are applied the email is distributed normally.

Copy Actions

This action is perform at a very early stage, before any other actions that could modify the email.

Copy functions are added READERS fields when they drop the message into the defined Notes databases. To have access you need to be granted with the [Admin] role or you have to either the Sender or one of the Recipients.

With the Journalling function, if the operation fails (overquota, database corrupt, or does not exists..) the email will remain in HOLD. The Filter engine will try again on its own.

Copy/Journalling Filter Rules UI :

[- Automatic copy, kill and/or reprocess -]		[- Comments / Warning, additional parameters -]	
Copy email to :		Include Doc Link :	<input checked="" type="checkbox"/> No
Move To Folder :		Keep ReturnReceipt :	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Journal into Mail-in Db :		Default view :	

You can as the email get copied move it into a folder. The folder will be created if necessary.
To define a folder with a hierarchy, use the following syntax :

- A\B\C

Where C is a sub-folder of B, itself a sub-folder of A

Keep ReturnReceipt : Y/N to remove or leave the Return Receipt field from the copied email.



Remove & Detach Actions

Removing actions are only impacting the current (FROM*TO(i)) virtual message. There is always an intact version of the email into the system.

The UI below can let you remove fields, attachments filtered by extensions or based on TEXT LIST field values, like what is performed by the AntiVirus which lists the files that contain virus.

Can also define a directory where to detach attachments, again filtered by extension.

Delete email	: <input type="radio"/> Yes <input checked="" type="radio"/> No		
Detach attachments	: <input checked="" type="radio"/> Yes <input type="radio"/> No	Detach directory Include host file link	: <input type="text"/> - Extension : <input type="text"/>
Remove attachments	: <input checked="" type="radio"/> Yes <input type="radio"/> No	Extension to remove (or) Field as List of file	: <input type="text"/>
Remove Fields	: <input type="text"/>		

Routing Actions

If you set "No" to 'Deliver Email To', the TO(i) will be retained into the final Recipients list.

NB : Fields SendTo, CopyTo, BlindCopyTo are unchanged

[-- Check addresses / add new recipients --]		[-- New routing conditions --]	
Deliver email to > (To)	: <input type="radio"/> Yes <input checked="" type="radio"/> No	Deliver email to > (To) vote	: [<input type="text"/> Majority <input type="text"/>]
Return email to > (From)	: <input type="radio"/> Yes <input checked="" type="radio"/> No	Route thru Domain	: <input type="text"/>
Forward email to >	: <input type="text"/>	Switch Importance To Low	: <input type="radio"/> Yes <input checked="" type="radio"/> No

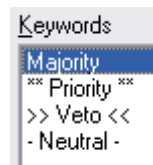
'Forward Email to' : Add some invisible recipients (user or group)

'Route thru Domain' : Change the way the router will route the email. Addresses are modified so they end up with a different domain : [abc@NouveauDomain](#). This is very efficient if you want some email to pass thru a certain server before reaching the final recipients.



Voting Notion

More than one rule can be applied to the same message. What happens if some rules are contradictory. By default an email is distributed to the final recipients, so each rule votes by default with a Majority value.



- Majority = 1
- Priority = $N (\text{Max} - \text{Rank})$
- Veto = Max
- Neutral = 0

The Max value is the number of rules.

The rank is the rule position into a sorting priority list, small rank is higher priority.

Example of 'Priority' value for a rule ranked 2 on 10(max) = $10 - 2 = 8$

if the rules vote (' Deliver Email to' = Yes) so the vote is positive : 8

if the rules vote (' Deliver Email to' = No) so the vote is negative : -8

When all the rules are done. If the final count is negative then the TO(i) is not retained into the final Recipients list.

All these “Blacklisted” addresses are kept into a memory to be used in case of special notification purpose.

Content Modifications Actions

The modification to the email can be configured with a RichText field and/or Text field. You can disable the modification without removing the content itself; for that just set the parameter : 'Append at the End of the Body' = No.

For the TEXT field. You can reuse the message content with the following syntax. Your text and {fieldName} will be appended to the original message body. To add new line use “ “ before carriage return.

One value is supplied by the filter engine; the {MW_TO} which stands for the current TO(i), from the {FROM*TO(i)} couples.

Append at the end of Body : ☒ Yes ☐ No



Append content (Rich Text) ...

If not possible use attachment: ...

Subject prefix : ...

Append to original mail 'Body' :

- MailWatcher Disclaimer -

Keep
your mail
ready
for business

Example of Content Modifications

Append at the end of Body : ☒ Yes ☐ No

Subject prefix : ...

Append RICH TEXT to original mail 'Body' :

Append TEXT supporting macro {FIELD} to original mail 'Body' :

test modification du body pour signaler à {FROM} que son mail {Subject} pour le destinaire {MW_TO}
a bien été reçu le {PostedDate}... voilà
saut de ligne

fin du message ici ->|

Supported fields : TEXT, TEXT LIST, NUMBER, DATE.



Example of received Email with Content Modifications

test modification du body pour signaler à **CN=admin notes_853_x64/O=mwatcher** que son mail **test sujet R22** pour le destinataire **CN=user test2/O=mwatcher@mwatcher** a bien été reçu le (YYYY/MM/DD) : 2012/12/14... voilà
saut de ligne

fin du message ici->

Notification Actions

This is one of the most powerful functionality of the Filter Engine. You can create a full email based on the original content and send it to the original sender, recipients, or others addresses based on parameters that can be static or dynamic

[..... Compose a new message]	
Send this notification/email : <input checked="" type="radio"/> Yes <input type="radio"/> No	
From address : <input]<="" td="" type="text" value="MailWatcher_"/> <td>Subject prefix : <input]<="" td="" type="text" value="Courrier non autorisé_"/></td>	Subject prefix : <input]<="" td="" type="text" value="Courrier non autorisé_"/>
Forward email to > : <input]<="" td="" type="text" value=""/> <td>Append the original mail 'Body' : <input type="radio"/> No <input checked="" type="radio"/> Bottom <input type="radio"/> Top</td>	Append the original mail 'Body' : <input type="radio"/> No <input checked="" type="radio"/> Bottom <input type="radio"/> Top
Deliver email to (From) : <input]<="" td="" type="text" value=""/> <td>Remove attachments : <input checked="" type="radio"/> Yes <input type="radio"/> No</td>	Remove attachments : <input checked="" type="radio"/> Yes <input type="radio"/> No
Deliver email to (To) : <input type="radio"/> Yes <input checked="" type="radio"/> No	
Message : <i>Votre message n'a pas été distribué. Vous n'êtes pas autorisé à utiliser cette liste de diffusion. Pour toute information prière de vous adresser à votre hiérarchie. _]</i>	

NB : By default the notification is aimed to the original sener. If you leave the 'Deliver email to' (From) empty, the notification will bounce back.

You can append the notification Body before or after the original body

- Append the original mail Body 'No', Bottom, Top.



Example of Notification with Info field and Dynamic addresses

[--- Compose a new message ---]	
Send this notification/email	<input checked="" type="radio"/> Yes <input type="radio"/> No
From address	: MailWatcher
Forward email to >	: NOHEADER
Deliver email to (From)	: {ReplyTo},{from}
Deliver email to (To)	: <input type="radio"/> Yes <input checked="" type="radio"/> No
Append Field Info	: <input checked="" type="radio"/> Yes <input type="radio"/> No
Subject prefix	: Delivery failure
Append the original mail 'Body'	: <input type="radio"/> No <input checked="" type="radio"/> Bottom <input type="radio"/> Top
Remove attachments	: <input checked="" type="radio"/> Yes <input type="radio"/> No
Field (values) appended as info	: \$UpdatedBy: From
Field (Titles) appended as info	: ---- Ceux qui ont fait des modifications ----; Sender

TEST MESSAGE NOTIFICATION RICH TEXT

- Forward Email To : {field1},{field2},DirectValue
- Deliver email to (FROM) : Leave this field empty or use {ReplyTo} for example.

At the bottom of the notification body you can add some extra information based on TEXT LIST field. You can configure a kind of report with the two following fields :

- Field (Values) appended as info : array values
- Field (Titles) Appended as info : array headers

Example of a received notification

TEST MESSAGE NOTIFICATION RICH TEXT

le corps

les PJ

dntrtr201203291153.out dntrtr201203291519.out dntrtr201203291530.out dntrtr201203291545.out dntrtr201204031233.out

Fin

test modification du body pour signaler à CN=admin notes_853_x64/O=mwatcher que son mail test sujet R21 a bien été reçu le (YYYY/MM/DD) : 2012/12/14... voilà
saut de ligne

fin du message ici ->

```
! ---- Ceux qui ont fait des modifications ---- ! Sender
!      CN=admin notes_853_x64/O=mwatcher      ! CN=admin notes_853_x64/O=mwatcher
! -----
```



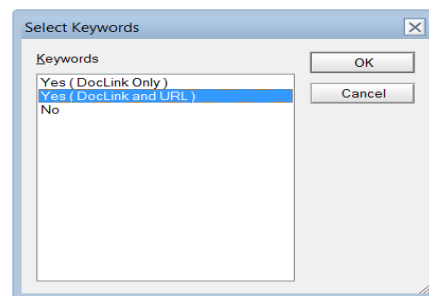
Combined Actions (Copy & notification)

You can combined the Copy actions with the Notification actions. While copying you can “Save” a Doclink to the copied document. If you set the “Include Doc Link” parameter then the notification will content this at the beginning of the notification body.

[-- Comments / Warning, additional parameters --]

Include Doc Link	: 『 Yes (DocLink Only) 』	Default view : 『 \$All 』
Keep ReturnReceipt	: <input checked="" type="radio"/> Yes <input type="radio"/> No	

You can set a DocLink and its URL equivalent too.





Les performances

The filter engine uses a cache to avoid to do too much query to the address book. Still groups can be modified while the engine is running.

So the cache is maintained up to date automatically. It keeps in memory the rules context. It does not keep the actions that can be modified on the fly.

- FROM-TO-FORMULA
- Group contents : FROM, TO, Except_From, Except_To.

Each MWADVT refreshed it saves the members list into a text file : **DumpBlackList.txt**

Dump/Cache file format

[Addresse or alias (shortname, internet adresse..)]
Group1, Group2,...

By default the engine refreshes its cache every 180 minutes. You can modified these values with the following notes.ini entry

MWREFRESHVIEWMODE = n (minutes)
ou
MWREFRESHVIEWMODE=never

To not to wait before a rules get in action, or if you change a rules context (FROM,TO,FORMULA,EXCEPT) you can run the following commands at the server console

```
> TELL MWADVT QUIT  
> LOAD MWADVT
```

Beware that Domino has its own cache you may need to do some > dbcach flush to force group updates.