

How to email Deltanji audit messages

05/03/2024 23:53:16

[FAQ Article Print](#)

| | | | |
|------------------|------------------------------------|---------------------|-----------------------|
| Category: | Products::Deltanji (formerly VC/m) | Votes: | 0 |
| State: | public (all) | Result: | 0.0 % |
| | | Last update: | 14:54:57 - 05/05/2016 |

Symptom (public)

Deltanji audit messages are being overlooked by users.

Problem (public)

Audit messages are only visible through the Deltanji UI. Messages can be flagged so they appear in the Audit pane whenever the flagged-for user logs in to the Deltanji browser UI, but notifications cannot be "pushed" by Deltanji.

Solution (public)

Use the "On Write" callout feature of Deltanji's Message Categories to run some custom code that will send emails.

Message categories are created and edited under the Setup folder. Transfer messages are written using whichever category is specified on the route definition. Change request messages use categories specified in Change Request Type maintenance. Other messages use categories specified in Setup\Properties\Audit Trail.

Below is a prototype subroutine. Put it into a site-specific routine that will be runnable from every Deltanji controlled namespace as well as from the DELTANJI or VCM namespaces itself (and the DELTANJI-LOCAL or VCM-LOCAL namespaces if you are using task server). Standard naming convention for such site-specific routines is to begin them %vcmz and follow this with your site name or abbreviation plus any other text you like, e.g. %vcmzAcmeCorpCallouts

```
EmailOnWrite(oAudit) ; onWrite callout for message categories
;
; Email the contents of the oAudit array using %Net.SMTP class
; Main useful subscripts of oAudit are:
; "message" - the audit message
; "objectVersion" - the obj the message relates to (null if message relates
to a CR)
; "dateTime" - the CCYYMMDDHHMMSS datetime of the message
; "user" - the Deltanji username generating the message
; Other subscripts:
; "changeRequest" - if the message relates to a CR rather than to an obj,
contains the CR, else null
; "object" - the base object name
; "variant" - the variant name
; "version" - the version number
; "category" - the audit message category
; "actionBy" - Deltanji username that the message is flagged for action by
; "actioned" - "no" if the action flag is set, else "yes"
; "transaction" - the transfer transaction that the message is associated
with
; "session" - the Deltanji user session in which the message was created
; "revision" - the revision counter of the message (increments when action
flag state or username changed)
;
new objMail,toAddresses,i,crlf,objSMTP,sc
set objMail=##class(%Net.MailMessage).%New()
set toAddresses="sysadmin@acmecorp.comZ" ; ***EDIT THIS (can be multiple
comma-separated addresses***
for i=1:1:(length(toAddresses," ")) do
objMail.To.Insert($piece(toAddresses," "))
set objMail.From="vcm@vcmserver" ; ***EDIT THIS***
set objMail.IsHTML=0
set objMail.Subject="Deltanji audit message"
set crlf=$c(13,10)
do objMail.TextData.Write("Deltanji has written an audit message." crlf)
do objMail.TextData.Write(crlf "message: " _ $g(oAudit("message")))
do objMail.TextData.Write(crlf "objectVersion: " _ $g(oAudit("objectVersion")))

do objMail.TextData.Write(crlf "dateTime: " _ $g(oAudit("dateTime")))
do objMail.TextData.Write(crlf "user: " _ $g(oAudit("user")))
do objMail.TextData.Write(crlf "changeRequest: " _ $g(oAudit("changeRequest")))

do objMail.TextData.Write(crlf "object: " _ $g(oAudit("object")))
do objMail.TextData.Write(crlf "version: " _ $g(oAudit("version")))
do objMail.TextData.Write(crlf "category: " _ $g(oAudit("category")))
do objMail.TextData.Write(crlf "actionBy: " _ $g(oAudit("actionBy")))
do objMail.TextData.Write(crlf "actioned: " _ $g(oAudit("actioned")))
do objMail.TextData.Write(crlf "transaction: " _ $g(oAudit("transaction")))
do objMail.TextData.Write(crlf "session: " _ $g(oAudit("session")))
do objMail.TextData.Write(crlf "revision: " _ $g(oAudit("revision")))
do objMail.TextData.Write(crlf "variant: " _ $g(oAudit("variant")))
do objMail.TextData.Write(crlf "<EOM>")
;
set objSMTP=##class(%Net.SMTP).%New()
```

```
set objSMTP.smtpserver="127.0.0.1" ; ***EDIT THIS***  
set sc=objSMTP.Send(objMail)  
quit
```

This will result in one email per audit message written with any message category you set the "On Write" field to "EmailOnWrite^%vcmzAcmeCorpCallouts". A more sophisticated solution might aggregate messages into a single email, perhaps grouping messages by their transaction number.