

Regex Use in Checks



CHECKCENTRAL



BINARYFORTRESS

Regex Use in Checks

Check configuration options provide a variety of approaches when matching incoming email messages and setting their status. This guide uses real-world examples to demonstrate check configurations using regex.

For Complex Match use in check configurations, see [Complex Match Use in Checks](#). For more basic check creation tips, see [Check Creation Guide and Best Practices](#).

Regex Examples

Set different statuses using number ranges

Source: ARQ email

View Message

Activity Info

Subject	ABC-ACCT1 ARQ (232 errors)
From	monitoring@everydaymsp.com
To	everydaymsp@mycheckcentral.cc
Originally To	monitoring@everydaymsp.com
Date	10/26/2022 12:24 AM
Check	ABC-ACCT1 (ARQ)

Message Info [Admin]

Message Content

Computer: ABC-ACCT1
Destination: Alderaan

Folder	Start Date	End Date	Scanned	Uploaded
C	10/26/2022 12:00 AM	10/26/2022 12:24 AM	345.148 GB	141 MB

Log:

Arq Agent version 5.15.1 started backup session for Alderaan/BBEE7529-55C2-48BF-80F3-EBA79C810D7B on 10/26/2022 12:00:04 AM

Backup session for C:\ started on 10/26/2022 12:00:05 AM

Found completed backup record for C:\

Error: C:\Users\everyday.ABC\AppData\Local\Microsoft\WindowsApps\GameBarElevatedFT_Alias.exe: Error opening

C:\Users\everyday.ABC\AppData\Local\Microsoft\WindowsApps\GameBarElevatedFT_Alias.exe: The file cannot be accessed by the system.

Error: C:\Users\everyday.ABC\AppData\Local\Microsoft\WindowsApps\MicrosoftEdge.exe: Error opening

C:\Users\everyday.ABC\AppData\Local\Microsoft\WindowsApps\MicrosoftEdge.exe: The file cannot be accessed by the system.

The subject line format of an ARQ email includes a number with the word "error" or "errors" (when there is more than 1 error), e.g. "(10 errors)." In this example we want to set a success status for errors of 100 or fewer and a warning status for any number of errors above 100.

To match on digits from 0 to 99 without going any higher, we'll need to include the opening parentheses "(" and the space after the number, along with the word "error." This regex looks like: `\(\d\d?\serror` Because the "error" text appears at the far right of our string, both "error" and "errors" will match.

What this *won't* match is when there are exactly 100 errors. For that, we'll add another rule looking for that exact phrase " (100 errors)." That is not a regex entry, and merely looks for that number letter phrase.

We use the "Any" configuration instead of "All," allowing for either entry to set the successful state.

Here's how it would look in the success criteria area of the check:

Success Criteria

Rules [+ Add Success Rule](#)

☒ Condense Whitespace

☒ Combine Attachments

Message must match Any of the following conditions:

Subject	Matches Regex	<input type="text" value="\d\d?\serror"/>		
Subject	Contains	<input type="text" value="(100 errors)"/>		

For the warning check, we require a minimum of 3 digits but allow for more. The regex will look like this:

By adding the plus symbol ("+") between the count and the space ("\s"), we open this to match much higher counts. We no longer need to match on single errors with the word "error," but the approach doesn't require any changes. Since 100 errors would match on this rule as well, we need to add an exclusion rule with the "Does Not Contain" option.

Because we need both rules to match, we use the "All" configuration.

Here's what both rules would look like in the warning area of the check:

Warning Criteria

Rules [+ Add Warning Rule](#)

☒ Condense Whitespace

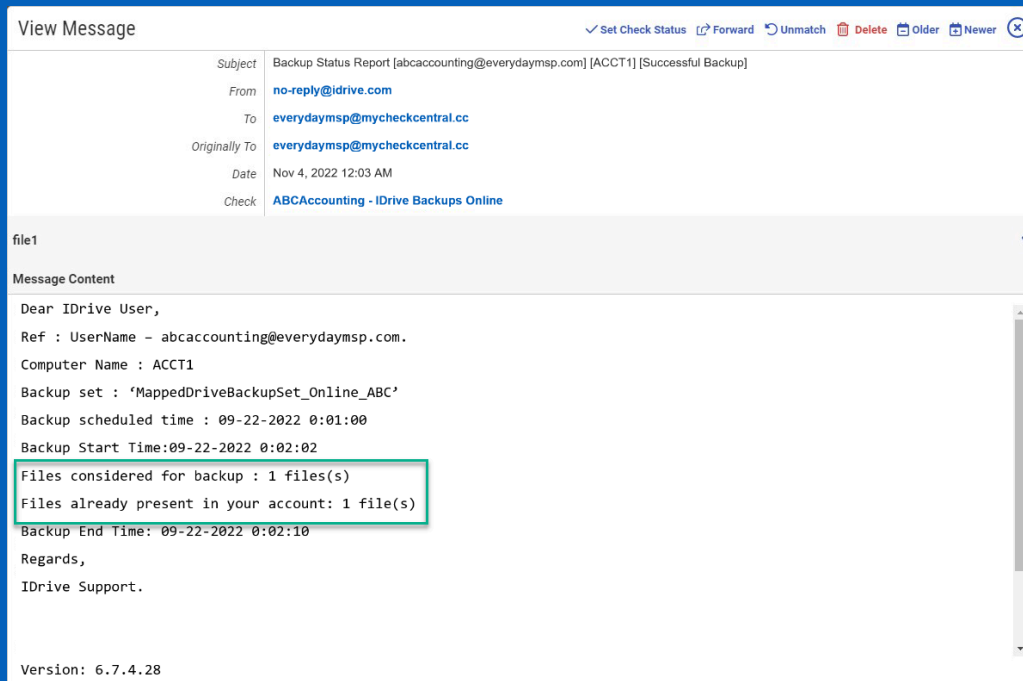
☒ Combine Attachments

Message must match All of the following conditions:

Subject	Matches Regex	<input type="text" value="\d\d\d+\serror"/>		
Subject	Does Not Contain	<input type="text" value="(100 errors)"/>		

Number of files is the same; no new files added to backup location

Source: IDrive



In this example we want a warning when no new files are added to a backup location. This is indicated in IDrive emails when the number of files considered for backup is identical to the number of files already present in the backup.

We're looking for an unspecified number of digits followed by the word "file," which would also include "file(s)." That's shown in regex like this: `(\d+ file)`

We then want to repeat the search for the same number and "file" combination later in the email. That's done with: `+\\1`

To make sure the correct repeated number/file combinations are being compared, we look for a character that doesn't appear between them, in this case a period, and use it as a breaking point. We do this by allowing anything else between our like number/file pair except a period, shown in regex as: `[^.]`

Any time our number/file pairs don't match, (the numbers are different), the warning status will be set.

Here's what it looks like in the warning area of our check:

Warning Criteria

Rules [+ Add Warning Rule](#)

- ☒ Condense Whitespace
- ☒ Combine Attachments

Message must match All of the following conditions:

Body Text Contains

Setting current status from a Datto success report which also reports past activity

Source: Datto success report, csv

```

File Edit Format View Help
"Client Name",Product,Application,"Services Fully Protected (24hrs)","Total Active Services","Never Fully Protected","Last Date Fully Protected",
"Aug 24, 2022","Aug 25, 2022","Aug 26, 2022","Aug 27, 2022","Aug 28, 2022","Aug 29, 2022","Aug 30, 2022","Aug 31, 2022","Sep 1, 2022",
"Sep 2, 2022"
"ABC Accounting",M365,Exchange,18,18,0,"Sep 1, 2022","Completed successfully","Completed successfully","Critical failure","Completed successfully",
"Completed successfully","Completed successfully","Completed successfully","Completed successfully","Completed successfully"
"Infinite Industries",M365,OneDrive,10,16,0,"Sep 1, 2022","Completed successfully","Completed successfully","Completed successfully","Completed successfully",
"Completed successfully","Completed successfully","Critical Failure"
"Ryan's Repairs",M365,Teams,6,6,0,"Sep 1, 2022","Completed successfully","Completed successfully","Critical failure","Completed successfully",
"Completed successfully","Completed successfully","Completed successfully","Completed successfully","Completed successfully"
"Southtown Services",M365,SharePoint,3,5,0,"Sep 1, 2022","Completed successfully","Completed successfully","Completed successfully","Completed successfully",
"Completed successfully","Completed successfully","Completed successfully","Completed successfully","Completed successfully","No data"
    
```

For this example, we want to set warning and failure statuses based on the most recent activity in a Datto success report (csv). These reports include previous activity as well as the most recent activity. The csv version of the report records each previous entry status followed by a comma (,). The last entry, which is the most recent activity, is not followed by a comma. Using this lack of comma focuses our search. The regex to exclude (allow all characters but a comma) the comma looks like this:

Failures are reported with the term "Critical Failure" while our warning status will report on values of "No data" or "Success with exception."

For our failure status, we combine the term and lack of immediately-following comma like this:

You can see what it looks like in the failure check area here:

Failure Criteria

Rules [+ Add Failure Rule](#)

☒ Condense Whitespace

☒ Combine Attachments

Message must match Any of the following conditions:

Attachments

Content Matches

"Critical failure"[^,]



For our warning status we'll use the terms "No data" and "exception" (not including the "Success with" portion of the text). These rules will also reference a lack of comma following the matching term, i.e.: `"No data" [^,]`

These rules also use the "Attachments" and "Content Matches Regex" options with the "Any" option, allowing for either rule to match.

You can see the warning area of the check here:

Warning Criteria

Rules [+ Add Warning Rule](#)

☒ Condense Whitespace

☒ Combine Attachments

Message must match Any of the following conditions:

Attachments

Content Matches

"No data"[^,]



Attachments

Content Matches

"exception" [^,]



Regex can be tested within CheckCentral by configuring your check and clicking the "Test" button (in the Save section when in wizard mode). This requires existing emails to test against, however. To test outside of CheckCentral, try <http://regexstorm.net/tester>

About CheckCentral

CheckCentral Monitoring consolidates and simplifies backup, system, and software email updates into a clean, graphical dashboard, bringing peace of mind to IT administrators of SMBs, Enterprises, and MSPs.

To learn more about CheckCentral, visit: <https://www.checkcentral.com>

About Binary Fortress Software

Binary Fortress has spent 17 years in pursuit of one goal: create software to make life easier. Our software ranges from display management and system enhancement utilities to monitoring tools and digital signage. IT administrators, professional gamers, coffee-shop owners, and MSPs all rely on Binary Fortress to make their days better, and their lives easier.

Copyright © 2007-2024 Binary Fortress Software, all rights reserved.

The Binary Fortress logo is a trademark of Binary Fortress Software.

The CheckCentral logo is a trademark of Binary Fortress Software.

Binary Fortress Software
1000 Innovation Drive, Suite 500
Kanata, Ontario, Canada
K2K3E7
<https://www.binaryfortress.com>