[<prev] [next>] [day] [month] [year] [list]

Date: Mon, 22 Feb 2021 17:58:40 +0100
From: Oswald Buddenhagen <oswald.buddenhagen@....de>
To: isync-devel@...ts.sourceforge.net
Cc: oss-security@...ts.openwall.com
Subject: CVE-2021-20247: isync/mbsync data leak/destruction vulnerability

description:

mbsync didn't validate the mailbox names returned by IMAP LIST/LSUB, which would allow a malicious/compromised server to use specially crafted mailbox names containing '..' path components to access data outside the designated mailbox on the opposite end of the synchronization channel. gory details follow below.
the attack vector is rather narrow, but the effects can be disastrous.
the vulnerability has been there "forever", though it wasn't of much concern prior to 1.3 used with a specific configuration.

mitigation:

upgrade to the freshly released v1.3.5 or v1.4.1 available from https://sourceforge.net/projects/isync/files/isync/ , or apply one of the attached patches (patches for earlier versions can be produced easily, should anyone care).

credit:

the possible existence of the vulnerability was suggested by a user who does not wish to be credited. :-D

vulnerability details:

- the victim must be using the Pattern channel option containing the '*' wildcard. this is fairly likely (even though only those who actually use hierarchical mailboxes (presumably a minority) actually need that - others may use the '%' wildcard).
- if the opposite end is also an IMAP server (which is presumed to be rare), the exact impact depends on the server. most servers will expose only a very restricted amount of data to any particular user, and will likely reject weird paths. also, most servers use '.' as the hierarchy delimiter, which would make mbsync reject the crafted paths as non-representable after delimiter translation. however, uw-imap/panda-imap for example would be vulnerable, as it basically just exposes the file system via IMAP.
- the much more common case is the opposite end being a local Maildir store, which is somewhat similar to the uw-imap case:
  - users who don't actually use hierarchical mailboxes locally are usually not vulnerable, as they won't set the SubFolders option, which will make mbsync reject any mailbox names containing hierarchy delimiters. (not applicable before v1.3.)
  - if SubFolders is Maildir++, no attack is possible, as periods are hierarchy delimiters and are consequently rejected by the translation. (not applicable before v1.3.)
  - if SubFolders is Legacy, an attack is limited to hidden directories, as a '.' is prepended to each subfolder when mapping to file system paths.
  - if SubFolders is Verbatim, exposure is unlimited. (not applicable before v1.3.)
  - the most likely target are Maildir folders that are synchronized to other servers (e.g., work vs. private mail stores)
    - if the victim is using '*' for the SyncState option (which most users seem to do judging by support requests; the example config file suggests it), all previously synchronized messages will be deleted from that folder (that this happens is a seperate bug that v1.4.1 also fixes), while new messages will be stolen.
    - otherwise, all messages from that folder will be stolen. i consider this the main danger of this vulnerability.
  - non-Maildir paths can be attacked only if they end with one of {cur,new,tmp}, as this is imposed by the expected Maildir structure.
    - if no 'cur' is present, the victim must have the Create option set for that end of the channel (this is likely).
    - all files from 'tmp' will be deleted
    - all files from 'cur' and 'new' will be stolen, and in the process renamed to add some Maildir "decorations"
    - subdirectories are not affected
    - in principle the attacker can deposit dangerous files, but these will also have "weird" names that cannot be influenced much, so they are unlikely to pose an actual threat. (general content attacks are neglected here, as they don't require this vulnerability to be executed.)
  - the attacked paths must be guessed or known in advance. if guessing is used and the victim has Create enabled, every attempted target path will be actually created, so the attacker will leave rather obvious traces, and might be even noticed before landing a single hit.
  - users who run mbsync interactively in verbose mode will likely spot an attack immediately (this is presumed to be rare; cron jobs are more likely).

i got an NVSS score of 'high', but there are lots of caveats that would qualify as mitigating factors if the criteria are not interpreted quite as literally (the case of a malicious server does not seem to fit very well).

View attachment "reject-funny-mailbox-names--1.3.patch" of type "text/x-diff" (2222 bytes)

View attachment "reject-funny-mailbox-names--1.4.patch" of type "text/x-diff" (2116 bytes)