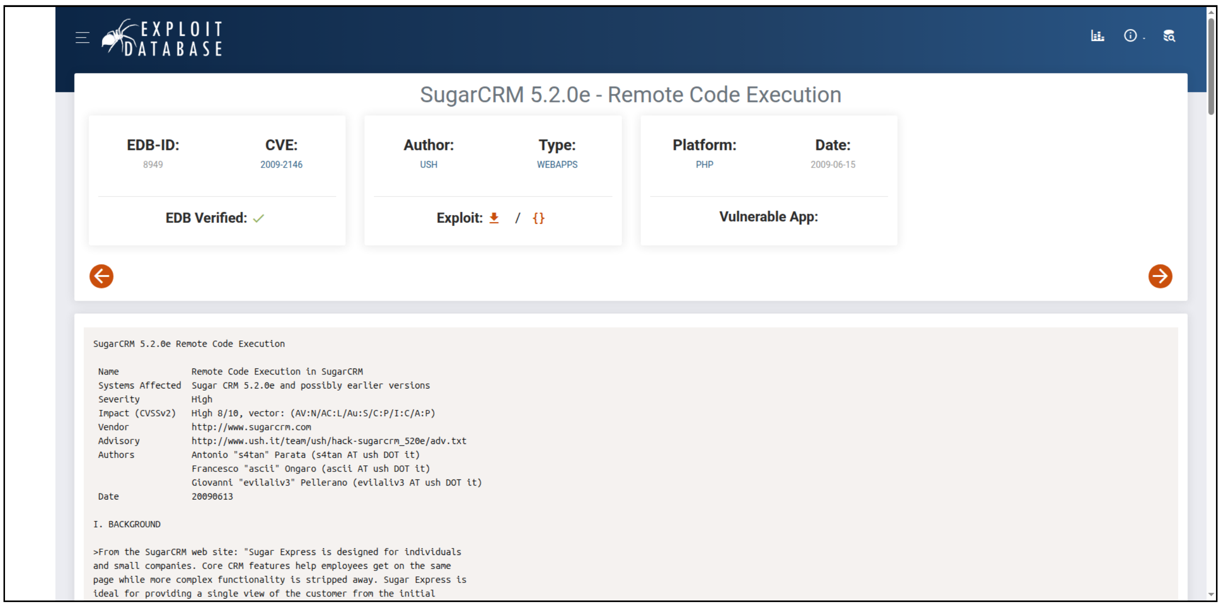
PRESS REVIEW ARCHIVE

Digital Media Monitoring & Documentation Service

Source URL:	https://www.exploit-db.com/exploits/8949
Archived Date:	August 15, 2025 at 15:01
Published:	June 15, 2009
Document Type:	Web Page Archive
Wayback Machine:	https://web.archive.org/web/*/https://www.exploit-db.com/exploits/8949

Page Screenshot



SugarCRM 5.2.0e - Remote Code Execution

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
8949	2009-2146	USH	WEBAPPS	PHP	2009-06-15

EDB Verified: ✓

Exploit: ⬇ / {}

Vulnerable App:

```
SugarCRM 5.2.0e Remote Code Execution

 Name              Remote Code Execution in SugarCRM
 Systems Affected  Sugar CRM 5.2.0e and possibly earlier versions
 Severity          High
 Impact (CVSSv2)   High 8/10, vector: (AV:N/AC:L/Au:S/C:P/I:C/A:P)
 Vendor            http://www.sugarcrm.com
 Advisory          http://www.ush.it/team/ush/hack-sugarcrm_520e/adv.txt
 Authors           Antonio "s4tan" Parata (s4tan AT ush DOT it)
                   Francesco "ascii" Ongaro (ascii AT ush DOT it)
                   Giovanni "evilaliv3" Pellerano (evilaliv3 AT ush DOT it)
 Date              20090613

I. BACKGROUND

>From the SugarCRM web site: "Sugar Express is designed for individuals
and small companies. Core CRM features help employees get on the same
page while more complex functionality is stripped away. Sugar Express is
ideal for providing a single view of the customer from the initial
```

SugarCRM 5.2.0e - Remote Code Execution

EDB-ID: 8949	**CVE:** 2009-2146

EDB Verified: ✓

Author: USH	**Type:** WEBAPPS

Exploit: ⬇ / {}

Platform: PHP	**Date:** 2009-06-15

Vulnerable App:

SugarCRM 5.2.0e Remote Code Execution

```
 Name              Remote Code Execution in SugarCRM
 Systems Affected  Sugar CRM 5.2.0e and possibly earlier versions
 Severity          High
 Impact (CVSSv2)   High 8/10, vector: (AV:N/AC:L/Au:S/C:P/I:C/A:P)
 Vendor            http://www.sugarcrm.com
 Advisory          http://www.ush.it/team/ush/hack-sugarcrm_520e/adv.txt
 Authors           Antonio "s4tan" Parata (s4tan AT ush DOT it)
                   Francesco "ascii" Ongaro (ascii AT ush DOT it)
                   Giovanni "evilaliv3" Pellerano (evilaliv3 AT ush DOT it)
 Date              20090613
```

I. BACKGROUND

>From the SugarCRM web site: "Sugar Express is designed for individuals
and small companies. Core CRM features help employees get on the same
page while more complex functionality is stripped away. Sugar Express is
ideal for providing a single view of the customer from the initial
marketing campaign through the sales cycle and on to customer support.
With Sugar Express, companies have a single system of truth for managing
customer interactions.".

II. DESCRIPTION

A Remote Code Execution Vulnerability exists in SugarCRM software.

III. ANALYSIS

Summary:

A Remote Code Execution issue has been found in SugarCRM version
5.2.0e. In order to exploit this vulnerability an account on the system
is required.

The vulnerability resides in the "Compose Email" section. The software
permits sending email with attachments (if not disabled by the
administrator). When the name of the file is specified, a validation
routine is called:

```
--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

function safeAttachmentName($filename) {
    global $sugar_config;
    $badExtension = false;
    //get position of last "." in file name
    $file_ext_beg = strrpos($filename, ".");
    $file_ext = "";
    //get file extension
    if($file_ext_beg > 0) {
        $file_ext = substr($filename, $file_ext_beg + 1);
    }
    //check to see if this is a file with extension located in "badext"
    foreach($sugar_config['upload_badext'] as $badExt) {
        if(strtolower($file_ext) == strtolower($badExt)) {
            //if found, then append with .txt and break out of lookup
            $filename = $filename . ".txt";
            $badExtension = true;
            break; // no need to look for more
        } // if
    } // foreach
    return $badExtension;
}

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--
```

This routine checks if the extension of the filename is blacklisted,
if so the ".txt" extension is appended to the filename. However there is
a coding error: the function assumes that the filename (extension
excluded) is at least one char long, this assumption is derived from the
statement:

```
--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

if($file_ext_beg > 0)

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--
```

Of course this is a bad assumption, if we set the whole filename to
".php" than the check is skipped and a void extension is assumed.
Because void extensions are not in the blacklist, no futher extension
is added to the filename. After this check a file is created on the
filesystem in the form "<id><filename>".

Where "id" is an alphanumeric string. With the trick illustrated we are
able to create a file with ".php" extension. To do this upload a new
file attachment and set the filename to ".php".

After this the attacker has to find the name of the file that was
uploaded in the attachment list files. To obtaint the real filename
look in the HTML response for a string like:

```
--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

<input value="6e25aba0-9dc4-2a57-8bae-4a1317b35d47.php" name="email_atta
chment0" id="email_attachment10" type="hidden">

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--
```

The real filename in this case is "6e25aba0-9dc4-2a57-8bae-4a1317b35d47.
php". Now the attacker has to find the directory where the file resides.

Again searching the HTML page for the attribute "assigned_user_id"
reveals the needed information:

```
--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

<a href="index.php?module=Emails&action=ListView&assigned_user_id=abf7c7
7b-2f71-8071-63ba-4a131068e9a2&type=archived">

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--
```

At this point the attacker has all the informations to invoke the
uploaded file.

Filename: 6e25aba0-9dc4-2a57-8bae-4a1317b35d47.php
Assigned user id: abf7c77b-2f71-8071-63ba-4a131068e9a2

To directly request it issue a request to:

```
--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

http://www.example.com/cache/modules/Emails/abf7c77b-2f71-8071-63ba-4a13
1068e9a2/6e25aba0-9dc4-2a57-8bae-4a1317b35d47.php

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--
```

As final note: if the user is "administrator", "assigned_user_id" is
always "1".

always "1".

IV. DETECTION

SugarCRM 5.2.0e and possibly earlier versions are vulnerable.

V. WORKAROUND

Upgrade to latest version 5.2.0f

VI. VENDOR RESPONSE

"We have fixed the issue and will be shipping the patch on June 12th.
We will be doing a full pass of quality assurance in this area to
ensure that no other issues crop up around file uploads.
The fix involves modifying the code that handles uploads for email
attachments to save the files using just a GUID rather than the original
file name. This is similar to how uploads are handled else where in the
application and should prevent the code from being executable on the
server side."

VII. CVE INFORMATION

No CVE at this time.

VIII. DISCLOSURE TIMELINE

```
20090519 Bug discovered
20090528 First vendor contact
20090528 Vendor Response
20090530 Vendor Confirm the vulnerability
20090602 Vendor propose a possible fix and path release
20090612 Vendor released SugarCRM 5.2.0f (Vulnerability fixed)
20090613 Advisory released
```

IX. CREDIT

Antonio "s4tan" Parata, Francesco "ascii" Ongaro and Giovanni
"evilaliv3" Pellerano are credited with the discovery of this
vulnerability.

Antonio "s4tan" Parata
web site: http://www.ush.it/
mail: s4tan AT ush DOT it

Francesco "ascii" Ongaro
web site: http://www.ush.it/
mail: ascii AT ush DOT it

Giovanni "evilaliv3" Pellerano
web site: http://www.ush.it/, http://www.evilaliv3.org/
mail: evilaliv3 AT ush DOT it

X. LEGAL NOTICES

Copyright (c) 2009 Francesco "ascii" Ongaro

Permission is granted for the redistribution of this alert
electronically. It may not be edited in any way without mine express
written consent. If you wish to reprint the whole or any
part of this alert in any other medium other than electronically,
please email me for permission.

Disclaimer: The information in the advisory is believed to be accurate
at the time of publishing based on currently available information. Use
of the information constitutes acceptance for use in an AS IS condition.
There are no warranties with regard to this information. Neither the
author nor the publisher accepts any liability for any direct, indirect,
or consequential loss or damage arising from use of, or reliance on,
this information.

# milw0rm.com [2009-06-15]

Tags: **Advisory/Source:** Link