

[UNIX] Pine URL Handler Allows Execution of Embedded Commands

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-01/0057.html>

From: support@securiteam.com

Date: 01/13/02

From: support@securiteam.com

To: list@securiteam.com

Date: Sun, 13 Jan 2002 23:55:15 +0100 (CET)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> -- Know that you're safe.

Pine URL Handler Allows Execution of Embedded Commands

SUMMARY

A security vulnerability in Pine allows a similar attack to <http://www.securiteam.com/unixfocus/5KPOP152AO.html> that noted in xchat. The vulnerability would allow remote attackers to cause the Pine program to execute arbitrary commands.

DETAILS

Vulnerable systems:

Pine 4.33 (under Redhat 7.0)

In Pine, if a user selects a URL with the structure of:

[http://address/'&/some/program\\${IFS}with\\${IFS}arguments&'](http://address/'&/some/program${IFS}with${IFS}arguments&')

And the URL handlers are installed, they will end up with the browser open on:

<http://address/>

And

</some/program> with arguments

Securiteam: [UNIX] Pine URL Handler Allows Execution of Embedded Commands

Will be executed.

If the attacked user is reading his email as root these commands will execute as root.

If you are reading your email as a non-privileged user, the impact is somewhat lower, although local exploits could be run on the computer, or Outlook style email viruses could be executed.

If you do not view links given to you in Pine, the impact from this problem is non-existent.

It is possible to obfuscate the URL by putting it in an HTML message such as the following.

Workaround:

It is recommended to create an alias for root to a non-privileged user instead of reading mail as root, to avoid root compromise using this, or similar, attacks.

Example:

-----Begin html email-----

From: Redhat Network Security <rhnsecurity@redhat.com>

To: undisclosed list <.@.>

Subject: Urgent update required to PINE

Message-ID: <Pine.LNX.4.33.0110221213510.9618-200000@clarity.local>

MIME-Version: 1.0

Content-Type: TEXT/html

Content-ID: <Pine.LNX.4.33.0110221214120.9618@clarity.local>

Content-Length: 389

Lines: 12

<HTML>

<BODY>

Urgent update:<p>

PINE allows execution of arbitrary commands.<p>

<a

href="http://updates.redhat.com/update_information/urgent/redhat-linux-version-7.0/hole-in-pine-url-handler/";to

http://updates.redhat.com/update_information/urgent/redhat-linux-version-7.0/hole-in-pine-url-handler/

<p>

This link contains PINE update information. <p>

You are advised to perform this immediately. <p>

The link also contains other urgent update information. <p>

</BODY>

</HTML>

-----End html email-----

Securiteam: [UNIX] Pine URL Handler Allows Execution of Embedded Commands

This would appear something like:

-----Begin view of email-----

Date: Mon, 22 Oct 2001 13:34:40 +1300
From: Redhat Network Security <rhnsecurity@redhat.com>
To: undisclosed list <.@.>
Subject: Urgent update required to PINE

Urgent update:

PINE allows execution of arbitrary commands.

http://updates.redhat.com/update_information/urgent/redhat-linux-version-7.0/hoec-in-pine-url-handler/

This link contains PINE update information.

You are advised to perform this immediately.

The link also contains other urgent update information.

-----End view of email-----

When this link is selected to follow, Pine changes the status/menu lines to read:

View selected URL

"http://updates.redhat.com/update_information/urgent/r..." ?

Y [Yes] U editURL

N No A editApp

This appears to match the URL in the email. This probably makes detection of this kind of exploit attempt harder.

Patch (SuSE):

--- pine/mailview.c.orig Thu Oct 12 21:33:32 2000

+++ pine/mailview.c Fri Oct 27 10:04:58 2000

@@ -3738,124 +3738,46 @@

#define URL_MAX_LAUNCH (2 * MAILTMPLN)

```
    if(handle->h.url.tool){
- char *toolp, *cmdp, *p, *q, cmd[URL_MAX_LAUNCH + 1];
- char *left_double_quote, *right_double_quote;
- int mode, len, hlen, quotable = 0, copied = 0, double_quoted = 0;
+ char *toolp, *cmdp, *endp, cmd[URL_MAX_LAUNCH + 1];
+ int mode, len, copied = 0;
        PIPE_S *syspipe;

        if((len = strlen(toolp = handle->h.url.tool)) > URL_MAX_LAUNCH)
            return(url_launch_too_long(rv));

- hlen = strlen(handle->h.url.path);
```

```

-
- /*
- * Figure out if we need to quote the URL. If there are shell
- * metacharacters in it we want to quote it, because we don't want
- * the shell to interpret them. However, if the user has already
- * quoted the URL in the command definition we don't want to quote
- * again. So, we try to see if there are a pair of unescaped
- * quotes surrounding _URL_ in the cmd.
- * If we quote when we shouldn't have, it'll cause it not to work.
- * If we don't quote when we should have, it's a possible security
- * problem (and it still won't work).
- *
- * In bash and ksh $( executes a command, so we use single quotes
- * instead of double quotes to do our quoting. If configured command
- * is double-quoted we change that to single quotes.
+ * Rather than trying to be smart about quoting and
+ * meta-characters, just stuff the URL into an environment
+ * variable and make the handler use it.
- */
-#ifdef _WINDOWS
- if(*toolp == '*' || (*toolp == '\"' && *(toolp+1) == '*'))
- quotable = 0; /* never quote */
- else
-#endif
- if(strpbrk(handle->h.url.path, "&*;<>?[]~$") != NULL){ /* specials? */
- if((p = strstr(toolp, "_URL_")) != NULL){ /* explicit arg? */
- int in_quote = 0;
-
- /* see whether or not it is already quoted */
-
- quotable = 1;
-
- for(q = toolp; q < p; q++)
- if(*q == '\"' && (q == toolp || q[-1] != '\\'))
- in_quote = 1 - in_quote;
-
- if(in_quote){
- for(q = p+5; *q; q++)
- if(*q == '\"' && q[-1] != '\\'){
- /* already single quoted, leave it alone */
- quotable = 0;
- break;
- }
- }
-
- if(quotable){
- in_quote = 0;
- for(q = toolp; q < p; q++)
- if(*q == '\"' && (q == toolp || q[-1] != '\\')){
- in_quote = 1 - in_quote;
- if(in_quote)

```

Securiteam: [UNIX] Pine URL Handler Allows Execution of Embedded Commands

```
- left_double_quote = q;
- }
-
- if(in_quote){
- for(q = p+5; *q; q++)
- if(*q == '\"' && q[-1] != '\\'){
- /* we'll replace double quotes with singles */
- double_quoted = 1;
- right_double_quote = q;
- break;
- }
- }
- }
- }
- else
- quotable = 1;
- }
- else
- quotable = 0;
+ setenv("URL", handle->h.url.path, 1);
+#define _URL_EXPANSION "\"$URL\""
```

```
    /* Build the command */
    cmdp = cmd;
- while(1)
- if(!*toolp && !copied)
- || (*toolp == '_' && !strncmp(toolp + 1, "_URL_", 4)){
+ endp = cmd + sizeof(cmd) - 1;
+ do {
+ if (cmdp + 1 > endp)
+ return(url_launch_too_long(rv));

+ if (!*toolp && !copied) {
    /* implicit _URL_ at end */
- if(!*toolp){
- *cmdp++ = ' ';
- len++;
- }
-
- /* add single quotes */
- if(quotable && !double_quoted){
- *cmdp++ = '\"';
- len += 2;
- }
+ *endp++ = ' ';
+ toolp = "_URL_";
+ }
+
+ if (strncmp(toolp, "_URL_", 5) != 0) {
+ *cmdp++ = *toolp++;
+ } else {
```

Securiteam: [UNIX] Pine URL Handler Allows Execution of Embedded Commands

```
+ toolp += 5; /* length of _URL_ */

- if((len += hlen) > URL_MAX_LAUNCH)
+ if (cmdp + sizeof(_URL_EXPANSION) - 1 > endp)
    return(url_launch_too_long(rv));

+ sstrncpy(&cmdp, _URL_EXPANSION);
    copied = 1;
- sstrncpy(&cmdp, handle->h.url.path);
- if(quotable && !double_quoted){
- *cmdp++ = "\";
- *cmdp = '\0';
- }
-
- if(*toolp)
- toolp += 5; /* length of "_URL_" */
- }
- else{
- /* replace double quotes with single quotes */
- if(double_quoted &&
- (toolp == left_double_quote || toolp == right_double_quote)){
- *cmdp++ = "\";
- toolp++;
- }
- else if(!(*cmdp++ = *toolp++))
- break;
    }
+ } while (*toolp);

    mode = PIPE_RESET | PIPE_USER ;
    if(syspipe = open_system_pipe(cmd, NULL, NULL, mode, 0)){
```

ADDITIONAL INFORMATION

The information has been provided by <<mailto:zen-parse@gmx.net>> zen-parse
and <<mailto:draht@suse.de>> Roman Drahtmueller.

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

list-unsubscribe@securiteam.com

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====

DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

- *Previous message:* support@securiteam.com: "[TOOL] A Simple Oracle Installation Security Scanner"
- *Messages sorted by:* [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#) [\[attachment \]](#)