

[NT] Multiple Vulnerabilities Found in Mailtraq (DoS, Password Decryption, Directory Traversal)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-06/0034.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 06/15/03

To: list@securiteam.com

Date: 15 Jun 2003 22:22:19 +0200

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

Latest attack techniques.

You're a pen tester, but is google.com still your R&D team?

Now you can get trustworthy commercial-grade exploits and the latest techniques from a world-class research group.

Learn more at <http://www.coresecurity.com/promos/site1>,
or call 617-399-6980

Multiple Vulnerabilities Found in Mailtraq (DoS, Password Decryption,
Directory Traversal)

SUMMARY

Mailtraq is a "comprehensive e-mail SMTP/POP3 and proxy server, with a powerful mailing list server". The product suffered from multiple vulnerabilities that range from access to files that reside outside the bounding HTML root directory (through denying access to the server by causing the server to utilize a high CPU percentage) through decryption of locally stored password, to a cross site scripting vulnerability in the web mail interface.

DETAILS

Vulnerable version:

* Mailtraq version 2.1.0.1302

Immune version:

Securiteam: [NT] Multiple Vulnerabilities Found in Mailtraq (DoS, Password Decryption, Directory Traversal)

* Mailtraq version 2.3.2.1419

HTTP Server directory traversal

By accessing a URL as simple as:

<http://127.0.0.1/win2k/>

Or,

<http://127.0.0.1/Program%20Files/>

It is possible to access directories that would be otherwise inaccessible. Some of the directories contain sensitive information, but what is more interesting in this problem is the fact that the Mailtraq product keeps the password encrypted in trivial form, which can be easily decrypted using the following perl script:

```
#!/usr/bin/perl
```

```
$Password = $ARGV[0];
```

```
print "Passwords should be something like: \\3D66656463626160\n";
```

```
print "Provided password: $Password\n";
```

```
$Password = substr($Password, 3);
```

```
$Length = length($Password)/2;
```

```
print "Length: $Length\n";
```

```
for ($i = 0; $i < $Length; $i++)
```

```
{
```

```
print "Decoding: ", substr($Password, $i*2, 2), " = ";
```

```
$ord = hex(substr($Password, $i*2, 2));
```

```
print $ord^$Length, " (", chr($ord^$Length), ")\n";
```

```
}
```

Note that it is possible to "decrypt" any password that is stored under the C:\Program Files\Mailtraq\database\configuration directory or under the users directory, both of which are accessible via the directory traversal vulnerability.

SMTP MAIL FROM, RCPT TO, HELO, FROM 100% CPU consumption (when viewing Event Log)

By sending a repeated a string such as @@%s%p%n, or without the @@ along with any of the SMTP commands, MAIL FROM, RCPT TO, HELO, email's FROM head field, will cause server's CPU usage to spike between 1 second to 5 seconds. Sending a simple overflow doesn't have the same effect. The number of repeated %s%p%n required in order to cause the DoS, is 65535 and above ("%s%p" x65535 – perl style).

Cross Site Scripting in WebMail

Sending a specially crafted email to a user can be used to steal his current session allowing an attacker to log on as the user. Sending such an email to the postmaster user will usually allow stealing of the

Securiteam: [NT] Multiple Vulnerabilities Found in Mailtraq (DoS, Password Decryption, Directory Traversal)

administrator session. The vulnerability occurs because the product does not correctly filter HTML/JavaScript code from the subject field when it is viewed in the ist (the email viewing itself is not vulnerable).

Example:

Sending an email with the following subject should illustrate the issue:

```
<script>alert(document.location)</script>
```

Logon CGI vulnerable to 100% CPU consumption

By sending an overly long username and password (any of them, or both) the CPU usage by the product will spike to 0%, the amount of time it spikes depends on the size of the buffer being sent (100,000 characters cause about 3–4 seconds stall)

```
POST /$/menu HTTP/1.1
```

```
Host:
```

```
User-Agent: Mozilla/1.0(compatible;)
```

```
Pragma: no-cache
```

```
Content-Length: ...depending on size...
```

```
Connection: close
```

```
Content-Type: application/x-www-form-urlencoded
```

```
user=<More than 100,000 A>&password=<More than 100,000 A>
```

Solution:

We recommend that all users upgrade to the most recent build of Mailtraq to ensure that they are up to date with the latest developments.

The latest build of Mailtraq Version 2.3.2.1419 includes the patches addressing these issues which are detailed above.

Mailtraq Version 2.3.2.1419 is immediately available for download as a public beta release pending complete QA testing, and then will be upgraded to full release status.

Vendor response:

HTTP Server directory traversal

Mailtraq is not vulnerable to this problem if it is installed with the default configuration on a standard "box". You can only access paths exposed by the web server.

Password Encryption

With respect to password encoding: weak password encryption was chosen as the objective is simply to obscure the information from the casual reader.

It is worth noting that by default .cfg files are excluded in the new Web Server.

SMTP MAIL FROM, RCPT TO, HELO, FROM 100% CPU consumption (when viewing Event Log)

We have investigated this issue and added constraints to the SMTP server.

Logon CGI vulnerable to 100% CPU consumption

These "vulnerabilities" only appear to exist when using the Event Log Viewer diagnostic-tool, not when Mailtraq is running in its normal configuration. However we have addressed the potential for high CPU consumption by capping the size of encoded POST data.

Under normal running conditions neither the Mailtraq Console or the event log viewer are open, so the "vulnerability" relies upon specific administrator activity.

Cross Site Scripting in WebMail

The example that you gave referred to the old and deprecated WebMail service. We recognise that this is a potentially significant issue and are grateful for your bringing it to our attention. It has been addressed in build 1419 which was released earlier today.

Mailtraq has replaced the entire WebMail system with a new one since the tested build. The new WebMail system was not susceptible to the problem you described, but CSS could be invoked in another manner. This has now been addressed.

It is important to note that the AUTHKEY cookie (allowing re-authentication after session expiry) is keyed to the client IP address. As of today's build, the same applies to the SESSIONKEY. Thus, even if a new CSS vulnerability were to arise, no useful information could be extracted from the browser.

The browse.asp* vulnerability which allows the attacker to determine the path of the installed web site has been addressed by limiting this debug information to the LAN specification.

We again thank you for bringing these items to our attention, and would be pleased to hear from you to discuss the matter further.

Best wishes,
David Rose

ADDITIONAL INFORMATION

The information has been provided by Noam Rathaus of <mailto:expert@securiteam.com> SecurITeam.

=====

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.