

[UNIX] Paros Proxy Blank Password

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2006-01/msg00016.html>

- *From:* SecuriTeam <support@xxxxxxxxxxxxxxxx>
 - *Date:* 4 Jan 2006 17:44:09 +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

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

Paros Proxy Blank Password

SUMMARY

<<http://www.parosproxy.org/>> Paros is "an intercepting HTTP/HTTPS proxy for use in security testing web applications".

Paros contains a flaw that allows a remote attacker to connect to a database port opened on the machine running Paros, without supplying any credentials.

DETAILS

Vulnerable Systems:

* Paros version 3.2.5 and below

The problem stems from use of a blank "sa" password on the open-source database ("HSQLDB") which is integrated with Paros.

The database server (which is written in Java) contains functionality for executing arbitrary Java statements. This is how HSQLDB provides Stored Procedure functionality.

The issue may result in disclosure of confidential data, and possible

[UNIX] Paros Proxy Blank Password

execution of commands on the victim machine.

A remote attacker may find credentials for web applications, valid session IDs, and confidential data downloaded from the website being tested with Paros. This information is present in the database.

Additionally, the possibility of executing Java statements on the database server may mean that an attacker can gain access to files or execute command at the OS level (by performing the Java equivalent of a "system()" call). This has not been investigated fully, but appears possible.

Disclosure Timeline:

03.10.05 – Problem discovered / reported

07.10.05 – Issue re-reported via sourceforge, as mail appeared lost in transit

07.10.05 – Paros developer releases updated version where DB listens on localhost only

Proof of concept:

To demonstrate this, first start Paros on the victim host (here, 192.168.0.1).

On the attacking host, ensure HSQLDB is installed, and add the following lines to the file \$HOME/sqltool.rc on the attacking host:

```
# connect to victimhost as sa, victimhost has IP 192.168.0.1
urlid victimhost-sa
url: jdbc:hsqldb:hsqldb://192.168.0.1
username sa
password
```

To connect using the "victimhost-sa" block above run:

```
java -jar $HSQLDB_HOME/jsqldb.jar victimhost-sa
```

At this point, it is possible to pull data from the tables in the database (browsing state, history, credentials).

The page at <http://hsqldb.org/doc/guide/ch09.html#call-section> also states it is possible to execute Java statements by writing them in the format "java.lang.Math.sqrt"(2.0).

ADDITIONAL INFORMATION

The information has been provided by <mailto:anc@xxxxxxxxxxxxxxxx> Andrew Christensen.

=====

[UNIX] Paros Proxy Blank Password

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@xxxxxxxxxxxxxxxx

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@xxxxxxxxxxxxxxxx

=====
=====

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.

-
- Prev by Date: [*\[EXPL\] MSDTC Arbitrary Opposite Memory Write Flaw \(Exploit\)*](#)
 - Next by Date: [*\[TOOL\] Arudius – Information Security Oriented Live CD Linux Distribution*](#)
 - Previous by thread: [*\[EXPL\] MSDTC Arbitrary Opposite Memory Write Flaw \(Exploit\)*](#)
 - Next by thread: [*\[TOOL\] Arudius – Information Security Oriented Live CD Linux Distribution*](#)
 - Index(es):
 - ◆ [*Date*](#)
 - ◆ [*Thread*](#)