

# [UNIX] Asterisk CallerID CDR SQL Injection

**Source:** <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-09/0044.html>

---

**From:** SecuriTeam ([support\\_at\\_securiteam.com](mailto:support_at_securiteam.com))

**Date:** 09/16/03

To: [list@securiteam.com](mailto:list@securiteam.com)

Date: 16 Sep 2003 15:37:08 +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>

-----

Asterisk CallerID CDR SQL Injection

---

## SUMMARY

<<http://www.asterisk.org/>> Asterisk is a complete PBX (Private Branch eXchange) in software. It runs on Linux and provides all of the features you would expect from a PBX and more. Asterisk does voice over IP with three protocols (SIP, IAX v1 and v2, and H323), and can interoperate with almost all standards-based telephony equipment using relatively inexpensive hardware.

Call Detail Records (CDRs) are generated by telephony systems in order to perform a number of functions such as billing and rating. CDRs contain a number of fields that identify useful information about the call including source, destination, and other items such as CallerID. These can be generated numerous times during the call to indicate the state of the call as well.

@stake found an issue while conducting a source code review of the CDR logging functionality. It is possible to perform SQL injection if an attacker can supply a malformed CallerID string.

The interesting thing to note about this vulnerability is that is cannot only be launched via VoIP protocols, but also through fixed-line connections (i.e. POTS – Plain Old Telephone System).

## Securiteam: [UNIX] Asterisk CallerID CDR SQL Injection

### DETAILS

@stake discovered that minimal input validation occurred between CDR generation and the acceptance of this data as part of the SQL query.

SQL injection is covered in details in:

1) SQL Injection –

<<http://www.spidynamics.com/papers/SQLInjectionWhitePaper.pdf>>  
<http://www.spidynamics.com/papers/SQLInjectionWhitePaper.pdf>

2) Advanced SQL Injection –

<[http://www.ngssoftware.com/papers/advanced\\_sql\\_injection.pdf](http://www.ngssoftware.com/papers/advanced_sql_injection.pdf)>  
[http://www.ngssoftware.com/papers/advanced\\_sql\\_injection.pdf](http://www.ngssoftware.com/papers/advanced_sql_injection.pdf)

As a result, it is possible for a remote unauthenticated user to perform arbitrary database operations.

Recommendation:

@stake notified the author of this particular code on the 17th of August. The author developed and deployed a patch silently to the CVS on the 9th of September.

@stake recommends that if you have not deployed a CVS version since the 9th of September 2003 to immediately do so.

### ADDITIONAL INFORMATION

The original advisory can be downloaded from:

<<http://www.atstake.com/research/advisories/2003/a091103-1.txt>>  
<http://www.atstake.com/research/advisories/2003/a091103-1.txt>.

The information has been provided by <<mailto:advisories@atstake.com>>  
@stake Advisories.

=====

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](mailto:list-unsubscribe@securiteam.com)

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