# CWE Detail – CWE-532

## Description

The product writes sensitive information to a log file.

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 3.0

Priority: P2 - Serious (High)

## Observed Examples (CVEs)

**•** CVE-2017-9615: verbose logging stores admin credentials in a world-readable log file

**•** CVE-2018-1999036: SSH password for private key stored in build log

## Related Attack Patterns (CAPEC)

* CAPEC-215

## Modes of Introduction

**•** Architecture and Design: COMMISSION: This weakness refers to an incorrect design related to an architectural security tactic.

**•** Implementation: N/A

**•** Operation: N/A

## Common Consequences

**•** Impact: Read Application Data — Notes: Logging sensitive user data, full path names, or system information often provides attackers with an additional, less-protected path to acquiring the information.

## Potential Mitigations

**•** Architecture and Design: Consider seriously the sensitivity of the information written into log files. Do not write secrets into the log files. (Effectiveness: N/A)

**•** Distribution: Remove debug log files before deploying the application into production. (Effectiveness: N/A)

**•** Operation: Protect log files against unauthorized read/write. (Effectiveness: N/A)

**•** Implementation: Adjust configurations appropriately when software is transitioned from a debug state to production. (Effectiveness: N/A)

## Demonstrative Examples

**•** N/A

**•** When the application encounters an exception it will write the user object to the log. Because the user object contains location information, the user's location is also written to the log.

**•** The error message that is created includes information about the database query that may contain sensitive information about the database or query logic. In this case, the error message will expose the table name and column names used in the database. This data could be used to simplify other attacks, such as SQL injection (CWE-89) to directly access the database.