# CWE Detail – CWE-836

## Description

The product records password hashes in a data store, receives a hash of a password from a client, and compares the supplied hash to the hash obtained from the data store.

## Extended Description

Some authentication mechanisms rely on the client to generate the hash for a password, possibly to reduce load on the server or avoid sending the password across the network. However, when the client is used to generate the hash, an attacker can bypass the authentication by obtaining a copy of the hash, e.g. by using SQL injection to compromise a database of authentication credentials, or by exploiting an information exposure. The attacker could then use a modified client to replay the stolen hash without having knowledge of the original password. As a result, the server-side comparison against a client-side hash does not provide any more security than the use of passwords without hashing.

## Threat-Mapped Scoring

Score: 3.0

Priority: P2 - Serious (High)

## Observed Examples (CVEs)

**•** CVE-2009-1283: Product performs authentication with user-supplied password hashes that can be obtained from a separate SQL injection vulnerability (CVE-2009-1282).

**•** CVE-2005-3435: Product allows attackers to bypass authentication by obtaining the password hash for another user and specifying the hash in the pwd argument.

## Related Attack Patterns (CAPEC)

* CAPEC-644
* CAPEC-652

## Attack TTPs

**•** T1558: Steal or Forge Kerberos Tickets (Tactics: credential-access)

**•** T1550.002: Pass the Hash (Tactics: defense-evasion, lateral-movement)

## Modes of Introduction

**•** Implementation: REALIZATION: This weakness is caused during implementation of an architectural security tactic.

## Common Consequences

**•** Impact: Bypass Protection Mechanism, Gain Privileges or Assume Identity — Notes: An attacker could bypass the authentication routine without knowing the original password.

## Applicable Platforms

**•** None (Class: Not Language-Specific, Prevalence: Undetermined)