# CWE Detail – CWE-297

## Description

The product communicates with a host that provides a certificate, but the product does not properly ensure that the certificate is actually associated with that host.

## Extended Description

Even if a certificate is well-formed, signed, and follows the chain of trust, it may simply be a valid certificate for a different site than the site that the product is interacting with. If the certificate's host-specific data is not properly checked - such as the Common Name (CN) in the Subject or the Subject Alternative Name (SAN) extension of an X.509 certificate - it may be possible for a redirection or spoofing attack to allow a malicious host with a valid certificate to provide data, impersonating a trusted host. In order to ensure data integrity, the certificate must be valid and it must pertain to the site that is being accessed. Even if the product attempts to check the hostname, it is still possible to incorrectly check the hostname. For example, attackers could create a certificate with a name that begins with a trusted name followed by a NUL byte, which could cause some string-based comparisons to only examine the portion that contains the trusted name. This weakness can occur even when the product uses Certificate Pinning, if the product does not verify the hostname at the time a certificate is pinned.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2012-5810: Mobile banking application does not verify hostname, leading to financial loss.

**•** CVE-2012-5811: Mobile application for printing documents does not verify hostname, allowing attackers to read sensitive documents.

**•** CVE-2012-5807: Software for electronic checking does not verify hostname, leading to financial loss.

**•** CVE-2012-3446: Cloud-support library written in Python uses incorrect regular expression when matching hostname.

**•** CVE-2009-2408: Web browser does not correctly handle '\0' character (NUL) in Common Name, allowing spoofing of https sites.

**•** CVE-2012-0867: Database program truncates the Common Name during hostname verification, allowing spoofing.

**•** CVE-2010-2074: Incorrect handling of '\0' character (NUL) in hostname verification allows spoofing.

**•** CVE-2009-4565: Mail server's incorrect handling of '\0' character (NUL) in hostname verification allows spoofing.

**•** CVE-2009-3767: LDAP server's incorrect handling of '\0' character (NUL) in hostname verification allows spoofing.

**•** CVE-2012-5806: Payment processing module does not verify hostname when connecting to PayPal using PHP fsockopen function.

**•** CVE-2012-2993: Smartphone device does not verify hostname, allowing spoofing of mail services.

**•** CVE-2012-5804: E-commerce module does not verify hostname when connecting to payment site.

**•** CVE-2012-5824: Chat application does not validate hostname, leading to loss of privacy.

**•** CVE-2012-5822: Application uses third-party library that does not validate hostname.

**•** CVE-2012-5819: Cloud storage management application does not validate hostname.

**•** CVE-2012-5817: Java library uses JSSE SSLSocket and SSLEngine classes, which do not verify the hostname.

**•** CVE-2012-5784: SOAP platform does not verify the hostname.

**•** CVE-2012-5782: PHP library for payments does not verify the hostname.

**•** CVE-2012-5780: Merchant SDK for payments does not verify the hostname.

**•** CVE-2003-0355: Web browser does not validate Common Name, allowing spoofing of https sites.

## Modes of Introduction

**•** Implementation: When the product uses certificate pinning, the developer might not properly validate all relevant components of the certificate before pinning the certificate. This can make it difficult or expensive to test after the pinning is complete.

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

## Common Consequences

**•** Impact: Gain Privileges or Assume Identity — Notes: The data read from the system vouched for by the certificate may not be from the expected system.

**•** Impact: Other — Notes: Trust afforded to the system in question - based on the malicious certificate - may allow for spoofing or redirection attacks.

## Potential Mitigations

**•** Architecture and Design: Fully check the hostname of the certificate and provide the user with adequate information about the nature of the problem and how to proceed. (Effectiveness: N/A)

**•** Implementation: If certificate pinning is being used, ensure that all relevant properties of the certificate are fully validated before the certificate is pinned, including the hostname. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** Even though the "verify" step returns X509\_V\_OK, this step does not include checking the Common Name against the name of the host. That is, there is no guarantee that the certificate is for the desired host. The SSL connection could have been established with a malicious host that provided a valid certificate.