# CWE Detail – CWE-624

## Description

The product uses a regular expression that either (1) contains an executable component with user-controlled inputs, or (2) allows a user to enable execution by inserting pattern modifiers.

## Extended Description

Case (2) is possible in the PHP preg\_replace() function, and possibly in other languages when a user-controlled input is inserted into a string that is later parsed as a regular expression.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2006-2059: Executable regexp in PHP by inserting "e" modifier into first argument to preg\_replace

**•** CVE-2005-3420: Executable regexp in PHP by inserting "e" modifier into first argument to preg\_replace

**•** CVE-2006-2878: Complex curly syntax inserted into the replacement argument to PHP preg\_replace(), which uses the "/e" modifier

**•** CVE-2006-2908: Function allows remote attackers to execute arbitrary PHP code via the username field, which is used in a preg\_replace function call with a /e (executable) modifier.

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Execute Unauthorized Code or Commands — Notes:

## Potential Mitigations

**•** Implementation: The regular expression feature in some languages allows inputs to be quoted or escaped before insertion, such as \Q and \E in Perl. (Effectiveness: N/A)

## Applicable Platforms

**•** PHP (Class: None, Prevalence: Undetermined)

**•** Perl (Class: None, Prevalence: Undetermined)

## Notes

**•** Research Gap: Under-studied. The existing PHP reports are limited to highly skilled researchers, but there are few examples for other languages. It is suspected that this is under-reported for all languages. Usability factors might make it more prevalent in PHP, but this theory has not been investigated.