# CWE Detail – CWE-288

## Description

The product requires authentication, but the product has an alternate path or channel that does not require authentication.

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2000-1179: Router allows remote attackers to read system logs without authentication by directly connecting to the login screen and typing certain control characters.

**•** CVE-1999-1454: Attackers with physical access to the machine may bypass the password prompt by pressing the ESC (Escape) key.

**•** CVE-1999-1077: OS allows local attackers to bypass the password protection of idled sessions via the programmer's switch or CMD-PWR keyboard sequence, which brings up a debugger that the attacker can use to disable the lock.

**•** CVE-2003-0304: Direct request of installation file allows attacker to create administrator accounts.

**•** CVE-2002-0870: Attackers may gain additional privileges by directly requesting the web management URL.

**•** CVE-2002-0066: Bypass authentication via direct request to named pipe.

**•** CVE-2003-1035: User can avoid lockouts by using an API instead of the GUI to conduct brute force password guessing.

## Related Attack Patterns (CAPEC)

* CAPEC-127
* CAPEC-665

## Attack TTPs

**•** T1542.002: Component Firmware (Tactics: persistence, defense-evasion)

**•** T1083: File and Directory Discovery (Tactics: discovery)

**•** T1556: Modify Authentication Process (Tactics: credential-access, defense-evasion, persistence)

**•** T1211: Exploitation for Defense Evasion (Tactics: defense-evasion)

## Modes of Introduction

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

**•** Architecture and Design: This is often seen in web applications that assume that access to a particular CGI program can only be obtained through a "front" screen, when the supporting programs are directly accessible. But this problem is not just in web apps.

## Common Consequences

**•** Impact: Bypass Protection Mechanism — Notes:

## Potential Mitigations

**•** Architecture and Design: Funnel all access through a single choke point to simplify how users can access a resource. For every access, perform a check to determine if the user has permissions to access the resource. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** The bugged line of code is repeated in the Bad
 example above. Weakness arises from the fact that the
 SECURE\_ME register can be modified by writing to the
 shadow register COPY\_OF\_SECURE\_ME, the address of
 COPY\_OF\_SECURE\_ME should also be included in the check.
 That buggy line of code should instead be replaced as
 shown in the Good Code Snippet below.

## Notes

**•** Relationship: overlaps Unprotected Alternate Channel