**Often Misused: Authentication Development Mitigation SOP**

Often misused vulnerabilities from authentication occur when attackers are able to spoof DNS entries. It is safe to assume your software will one day run in an environment with a compromised DNS server. If attackers are allowed to make DNS updates, it is possible to route your network traffic through their machines or make it appear as if their IP addresses are part of your domain. The security of your system should be not based on DNS names.

**Defense Against Often Misused: Authentication**

Check to make sure the host’s forward and backward DNS entries match because attackers will not be able to spoof the forward and reverse DNS entries without controlling the nameservers for the target domain. While no authentication mechanism is foolproof, multi-factor authentication that includes a physical token offers the most security available at a reasonable price.

**Example**

String ip = request.getRemoteAddr();

InetAddress addr = InetAddress.getByName(ip);

if(addr.getCanonicalHostName().endsWith(“trustme.com”)) {

trusted = true;

}

**Explanation**

The code above used a DNS lookup in order to decide whether or not an inbound request is from a trusted host, which could give an attacker trusted status if they are able to poison the DNS cache.

**References**

1. [HP Enterprise Security – Often Misused: Authentication](http://www.hpenterprisesecurity.com/vulncat/en/vulncat/dotnet/often_misused_authentication.html)
2. [OWASP – Often Misused: Authentication](https://www.owasp.org/index.php/Often_Misused:_Authentication)