# Nginx - All Prevention

**Prevention** secara singkat berarti **pencegahan** atau **tindakan untuk menghindari suatu risiko atau ancaman sebelum terjadi**.

Dalam konteks keamanan, **prevention** bertujuan untuk **melindungi sistem dari serangan, eksploitasi, atau penyalahgunaan** dengan menerapkan aturan atau mekanisme pengamanan

Berikut Konfigurasinya :

```
############################ XSS PREVENTION
##############################
        set $block_xss 0;
if ($query_string ~ "base64_(en|de)code\(.*\)") {
set $block_xss 1;
}
if ($request_uri ~ "base64_(en|de)code\(.*\)") {
        set $block_xss 1;
}
if ($query_string ~ "(<|%3C).*script.*(>|%3E)") {
        set $block_xss 1;
}
if ($request_uri ~ "(<|%3C).*script.*(>|%3E)") {
        set $block_xss 1;
}
if ($query_string ~ "(<|%3C).*iframe.*(>|%3E)") {
        set $block_xss 1;
}
if ($request_uri ~ "(<|%3C).*iframe.*(>|%3E)") {
        set $block_xss 1;
}
if ($query_string ~ "GLOBALS(=|\[|\%[0-9A-Z]{0,2})") {
        set $block_xss 1;
}
if ($query_string ~ "_REQUEST(=|\[|\%[0-9A-Z]{0,2})") {
        set $block_xss 1;
}
if ($block_xss = 1) {
```

```
        return 403;
}

## Block SQL injections
    set $block_sql_injections 0;
    if ($query_string ~ "union.*select.*\(") {
        set $block_sql_injections 1;
    }
    if ($query_string ~ "union.*all.*select.*") {
        set $block_sql_injections 1;
    }
    if ($query_string ~ "concat.*\(") {
        set $block_sql_injections 1;
    }
    if ($block_sql_injections = 1) {
        return 403;
    }

    ## Block file injections
    set $block_file_injections 0;
    if ($query_string ~ "[a-zA-Z0-9_]=http://") {
        set $block_file_injections 1;
    }
    if ($query_string ~ "[a-zA-Z0-9_]=(\.\.//?)+") {
        set $block_file_injections 1;
    }
    if ($query_string ~ "[a-zA-Z0-9_]=/([a-z0-9_.]//?)+") {
        set $block_file_injections 1;
    }
    if ($block_file_injections = 1) {
        return 403;
    }

    ## Block common exploits
    set $block_common_exploits 0;
    if ($query_string ~ "(<|%3C).*script.*(>|%3E)") {
        set $block_common_exploits 1;
    }
    if ($query_string ~ "GLOBALS(=|\[|\%[0-9A-Z]{0,2})") {
        set $block_common_exploits 1;
```

```
}
if ($query_string ~ "_REQUEST(=|\[|\%[0-9A-Z]{0,2})") {
    set $block_common_exploits 1;
}
if ($query_string ~ "proc/self/environ") {
    set $block_common_exploits 1;
}
if ($query_string ~ "mosConfig_[a-zA-Z_]{1,21}(=|\%3D)") {
    set $block_common_exploits 1;
}
if ($query_string ~ "base64_(en|de)code\(.*\)") {
    set $block_common_exploits 1;
}
if ($block_common_exploits = 1) {
    return 403;
}

## Block spam
set $block_spam 0;
if ($query_string ~ "\b(ultram|unicauca|valium|viagra|vicodin|xanax|ypxaieo)\b") {
    set $block_spam 1;
}
if ($query_string ~ "\b(erections|hoodia|huronriveracres|impotence|levitra|libido)\b") {
    set $block_spam 1;
}
if ($query_string ~ "\b(ambien|blue\spill|cialis|cocaine|ejaculation|erectile)\b") {
    set $block_spam 1;
}
if ($query_string ~ "\b(lipitor|phentermin|pro[sz]ac|sandyauer|tramadol|troyhamby)\b") {
    set $block_spam 1;
}
if ($block_spam = 1) {
    return 403;
}

## Block user agents
#SET $BLOCK_USER_AGENTS 0;

# Don't disable wget if you need it to run cron jobs!
#if ($http_user_agent ~ "Wget") {
```

```
#    set $block_user_agents 1;
#}

# Disable Akeeba Remote Control 2.5 and earlier
if ($http_user_agent ~ "Indy Library") {
    set $block_user_agents 1;
}

# Common bandwidth hoggers and hacking tools.
if ($http_user_agent ~ "libwww-perl") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "GetRight") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "GetWeb!") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "Go!Zilla") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "Download Demon") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "Go-Ahead-Got-It") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "TurnitinBot") {
    set $block_user_agents 1;
}
if ($http_user_agent ~ "GrabNet") {
    set $block_user_agents 1;
}

if ($block_user_agents = 1) {
    return 403;
}
```

Revision #6
Created 7 July 2024 01:46:32 by Kevin

Updated 19 February 2025 04:38:06 by Kevin