INTISARI

Pada penelitian ini akan dibahas mengenai permasalahan implementasi keamanan pada *web server* khususnya mengenai tidak adanya panduan atau *template* dalam konfigurasi keamanan *web server* dan adanya kerentanan pada *web* yang digunakan. Permasalahan pertama adalah tidak adanya panduan atau *template* dalam konfigurasi keamanan ditengah penggunaan teknlogi informasi berupa situs *web* pada *web server* oleh sumber daya manusia yang bukan dari bidang teknologi informasi. Permasalahan tersebut menjadi alasan diperlukannya panduan atau *template* konfigurasi keamanan *web server*. Permasalahan kedua adalah ditemukannya kerentanan *SQL injection* pada situs *web* dari BPPTKG. Kerentanan ini menyebabkan *hacker* dapat memperoleh data dari *database* tanpa menggunakan kredensial khusus sehingga mempengaruhi kerahasiaan dari *database* yang digunakan bahkan dapat mempengaruhi integritas apabila ada miskonfigurasi hak akses di *database*.

Dalam memberikan solusi tidak adanya panduan atau *template* konfigurasi, peneliti akan membuat sebuah *template* konfigurasi yang dibuat dari *ansible,* sebagai *tool* untuk *deployment*, yang terdiri dari berbagai rekomendasi terbaik konfigurasi dari *CIS* (*Center for Internet Security*). *Template* tersebut dapat dijalankan dengan sekali eksekusi sehingga mempermudah dalam melakukan konfigurasi karena menghemat waktu lebih banyak dibandingkan melakukan satu persatu konfigurasi secara manual. Dalam memberikan solusi penanganan kerentanan *SQL injection,* peneliti membuat *script* deteksi dan mitigasi kerentanan secara *static* yaitu dengan langsung mendeteksi adanya kerentanan di kode *PHP* dan memberikan mitigasinya langsung di kode *PHP*.

Hasil dari eksekusi *template* konfigurasi keamanan *web server* adalah *template* konfigurasi yang dibuat dengan *ansible* dapat membandingkan konfigurasi saat ini di *web server* dengan rekomendasi yang dituliskan di *template* dan melakukan perubahan apabila tidak sesuai. Hasil konfigurasi yang dilakukan juga tidak menimbulkan kendala pada *web server.* Sedangkan hasil eksekusi *script* tersebut peneliti berhasil melakukan mitigasi kerentanan *SQL injection* yang sifatnya masih cakupan global (selain antara fungsi dan antara *class* yang berbeda).

Kata kunci : *Ansible, CIS* (*Center for Internet Security*), *PHP, SQL Injection, Web Server*

ABSTRACT

At this final project will be discussed problem about how to analyze and implement security at Web Server especially about an absence of a guide or template for security configuration at web server and a presence of security vulnerability at web site. The first problem that will be discussed is the absence of guide or security configuration template at web server in the midst of using information technology in the form of website at web server by human resources who are not from information technology field. This problem is the reason for the need of a web server security configuration guide or template. The second problem is the discovery of SQL injection vulnerabilities on the website of BPPTKG. This vulnerabilities can cause hacker to get data from database without using special credential so it affects the confidentiality of the database and can even affect integrity if misconfiguration of access rights presence in the database

For giving a solution about two problems before, the writer will create a configuration template by using Ansible, as a tool for deployment, which consist of various configuration suggestions from CIS (Center for Internet Security) that can be ran with one time execution so it will ease an administrator and reduce configuration time in comparison with configuration time when it is ran one by one. While for the second problem about the security vulnerability which is SQL Injection that found by the past pentester at BPPTKG, the solution that proposed by a writer for the second problem is creating a detection and mitigation script to detect and mitigate SQL Injection statically at PHP code.

The result from executing security configuration template at web server made with ansible is that it can compare present configuration at web server with configuration recommendation that written at configuration template and change configuration if there is difference. The result of configuration that have been done also did not cause a problem at web server. At the result of execution of that script, writer sucessfully mitigate the security vulnerability which is at global scope (beside interfunction and interclass).

Keywords : Ansible, CIS (Center for Internet Security), PHP, SQL Injection, Web Server