BAB I

PENDAHULUAN

1.1 Latar Belakang

Kasus peretasan suatu website bukan merupakan hal yang asing lagi untuk saat ini. Hilangnya data dan hilangnya kepercayaan user merupakan dua hal dari sebagian kerugian yang timbul akibat adanya peretasan. Suatu insiden peretasan pada server tidak bisa dibebankan langsung pada satu pihak yang menangani website, dikarenakan ada beberapa pihak yang secara langsung dapat ikut andil dalam usaha pengamanan website. Dilihat dari sisi industri saat ini, perusahaan membedakan penugasan terhadap tenaga kerja menjadi dua bagian. Yang pertama adalah *webmaster* atau *website administrator* yang bertugas mengolah konten yang ada pada aplikasi website. Yang kedua adalah *sysadmin* atau *sistem administrator* yang bertugas melakukan monitoring dan maintenance dari sisi server. Kedua bagian tersebut mempunyai pernanan masing – masing dan bisa saling berkolaborasi untuk menciptakan suatu website yang aman.

ModSecurity bekerja sebagai filter http query berdasarkan rule tertentu dan diterapkan pada sisi server. Dengan mengimplementasikan *ModSecurity* sebagai filter maka berbagai jenis kemungkinan isu keamanan dapat dicegah secara dini. Namun penerapan *ModSecurity* di dalam sebuah web server tidak seutuhnya

menjamin keamanan website. Karena rule atau filter perlu dibuat secara spesifik terhadap website yang ada di dalam server.

Perkembangan metode peretasan sejalan dengan perkembangan di sisi keamanan. Di mana terdapat patch untuk menutup cara tertentu, maka akan lahir cara baru yang digunakan oleh peretas. Misalnya, setelah dibuatkan rule untuk menangani beberapa jenis metode injeksi yang populer seperti *cross-site scripting (XSS), file inclusion* dan *SQL injection* muncul metode baru yang belum terbuat rule untuk metode tersebut, contohnya adalah *file upload, php code injection dan php object injection*. Kelemahan seperti ini biasanya ditemukan di dalam website dengan *CMS (Content Management System)* yang terinstall modul dari pihak ketiga. Oleh karena itu diperlukan perhatian khusus dalam hal pengamanan suatu website dari sisi server untuk dapat menyediakan keamanan yang optimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas ada beberapa masalah yang dapat dirumuskan, di antaranya :

1. Bagaimana implementasi web application firewall (modsecurity) pada web server apache ?
2. Bagaimana langkah membuat filter pada *ModSecurity* berdasarkan informasi kelemahan yang didapat ?
3. Bagaimana langkah membuat filter dengan memanfaatkan data log yang ada di dalam server ?

4. Apakah implementasi web application firewall dapat menangani masalah yang ada ?
5. Apakah keunggulan penerapan mod security dibandingkan dengan penanganan keamanan dengan cara lain ?

1.3 Ruang Lingkup

Seperti yang telah diketahui, isu keamanan adalah hal yang tidak bisa diprediksi. Oleh karena itu untuk membatasi pelebaran dari penelitian dan beberapa pertimbangan lainnya, maka diuraikan beberapa hal berikut ini :

1. Penelitian ini mengambil sudut pandang dari seorang system administrator dan bukan dari sisi developer atau wesite administrator.
2. Simulasi jaringan dari web server yang digunakan pada penelitian merupakan jaringan virtual pada sebuah laptop.
3. Implementasi web application firewall (modsecurity) dilakukan pada web server apache.
4. Pengujian dibatasi untuk kelemahan yang terdapat pada *cms* yang telah ditentukan yaitu *wordpress* dan *joomla*.
5. Jenis kelemahan yang akan diteliti adalah :
    a. File Upload
        - Wordpress Plugin Mac Photo Gallery 2.7
        - Wordpress Plugin Asset Manager 0.2
        - Wordpress Plugin AllWebMenus < 1.1.9
        - Wordpress Plugin LearnDash 2.5.3

b. Code Injection

- Wordpress Plugin Insert PHP 3.3.1
- Wordpress Plugin W3 Total Cache 0.9.2.3
- Joomla! component com_civicrm 4.2.2

c. PHP Object Injection

- Wordpress Plugin Ultimate Product Catalog <= 4.2.24
- Joomla! 1.5 < 3.4.5 - Object Injection 'x-forwarded-for'
- Joomla! 3.0.2 - 'highlight.php' PHP Object Injection
- Joomla! 3.0.3 - 'remember.php' PHP Object Injection

6. Contoh isu keamanan yang digunakan merupakan isu yang sudah ada di ranah publik.
7. Malware yang akan digunakan merupakan malware dari jenis phpshell yang umum dan tersedia di internet sebagai contoh implementasi pada tahap maintenance server.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu :

1. Menghasilkan sebuah alternatif sistem keamanan dari sisi web server yang dapat digunakan.
2. Melakukan upaya pembuatan sistem keamanan untuk pencegahan terjadinya peretasan.
3. Memberikan alternatif kepada system administrator yang dapat digunakan untuk membantu dalam proses maintenance server yang telah terinfeksi malware.

4. Menerapkan pembuatan rule mod security secara dinamis menyesuaikan lingkungan server dan website yang digunakan.
5. Menerapkan pembuatan rule mod security secara dinamis berdasarkan dugaan isu keamanan di dalam usaha pengamanan server yang berlangsung secara terus – menerus.

1.5 Manfaat Penelitian

Beberapa manfaat yang dapat diuraikan dengan diadakannya penelitian ini adalah seperti berikut :

1. Memberikan gambaran beberapa metode intrusi sehingga developer aplikasi web diharapkan menjadi lebih perhatian terhadap keamanan dalam proses pembuatan website.
2. Pemahaman tentang beberapa metode peretasan pada aplikasi web dan penanganannya dengan menggunakan web application firewall.
3. Pemahaman pembuatan filter pada mod security pada suatu insiden atau informasi tentang kelemahan tertentu.
4. Implementasi pada penelitian ini dapat diterapkan pada perusahaan yang bergerak pada jasa web hosting sebagai langkah preventif dan menyediakan sistem keamanan untuk client.
5. Implementasi pada penelitian ini dapat pula diterapkan pada suatu instansi tertentu sebagai bentuk usaha pengamanan website menyesuaikan dengan lingkungan server yang ada.