SOFTWARE SOLUTION FOR VULNERABILITY DETECTION

Kahtan Hussein, Momoh Angelo, A.I. Bukshtynova

Web Application Architecture can be of various types. Web applications have been developed on the basis of improvement and complication of web-node functions. Web application extends functions of a web-node by making it possible for its clients to use business logic and hence to modify the data on the server. This definition of web applications specifies that it consists of at least three important architectural components: a client browser, a web server, and an applications server.

Often a web application also uses a database server. A more rigorous definition of a web application can be as follows: a client/server software system, comprising of at least the following architectural components: HTML/XML browser on one or more client computers, communicating with the web server through the HTTP protocol, and the applications server which manages the business logic.

To detect vulnerabilities in web applications, there are many methods, such as the penetration testing method, the source code analysis of the application. The penetration testing method is the most accessible, since it only requires access to the web application. And such access is available to all users, if the application is accessible from the Internet.

These methods include the following steps: passive information gathering; definition of the web environment and platform; port scanning / collection of service banners; automatic scanning; definition of «weak points» of the resource; manual analysis; collection and analysis of information received; analysis of attack vectors; confirmation of received vectors; compilation of a report.

To implement the software solution, the Python programming language was chosen. This choice is conditioned by the fact that all operating systems support this language and there are a great number of libraries to use. The software solution involves the library for the work with networks and sub networks of IPv4 (netaddr) protocol, the library for the processing of HTML-markup which helps to extract data (bs4), and the library for asynchronous processing of web-services (gevent), which helps to speed up the software performance through the concurrent execution of commands not sequential.

The program complex functions on following algorithm. On a program input the information on a target network moves. These data can be presented in the form of the domain of the second level, a network of IPv4 addresses, the list of domains of the third level, the list of IPv4 addresses. Depending on the data submitted on an input the file networkscanner forms the list of not repeating addresses. The list is transferred to a file portscanner.py which on an exit returns data about open ports and the office information on network sites and transfers these given to a file report.py. The file report.py processes the received results and deduces to their user.

СИМБИОЗ SIEM И DLP

Т.А. Андриянова, С.Б. Саломатин

В настоящее время одной из наиболее актуальных угроз в области информационной безопасности является утечка конфиденциальных данных от несанкционированных действий сотрудников. Согласно исследованиям за прошлый год, в более чем 80 % случаев утечка информации в компаниях происходила по вине (или неосторожности) именно внутреннего нарушителя. Таким образом, DLP технологии становятся неотъемлемой и обязательной частью корпоративных систем информационной безопасности, существенно повышая уровень защиты конфиденциальных данных. Это обусловлено еще и тем, что большая часть традиционных средств защиты, таких как антивирусы, межсетевые экраны и системы аутентификации не способны обеспечить эффективную защиту от внутренних нарушителей.

Одним из наиболее перспективных вариантов расширения функционала DLP систем является интеграция с SIEM технологией. В симбиозе системы взаимно дополняют друг друга. Компоненты DLP-системы осуществляют поиск и классификацию защищаемой информации по установленным критериям. А SIEM формирует «единое окно» для администратора безопасности, в котором сводятся данные о выявленных файлах, подлежащих защите, попытках доступа к ним, а также коррелируется технологическая информация, поступающая от ОС, СУБД, сетевого оборудования и других источников, формируя полную картину состояния информационной безопасности в организации.