МЕТОДЫ И СРЕДСТВА БЕЗОПАСНОГО ИСПОЛЬЗОВАНИЯ ПОПУЛЯРНЫХ СИСТЕМ УПРАВЛЕНИЯ СОДЕРЖИМЫМ ВЕБ-САЙТОВ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Цалко А. С.

Кучинский П. В. – д-р. физ.-мат. наук, профессор

С каждым годом растет доля сайтов, использующих широко распространенные системы управления содержимым (CMS). В 2016 году из 10 миллионов крупнейших веб-сайтов 26.2% работают на CMS WordPress. Суммарно же доля данной системы составляет 59.3% среди всех используемых CMS [1]. Широкое распространение как данной, так и других популярных систем делает их мишенью для злоумышленников.

Как правило, использование самих CMS не несет угрозы. Однако в декабре 2015 года была найдена критическая уязвимость в системе «Joomla», затрагивающая все использующие данную CMS сайты (около 3 миллионов веб-сайтов) [2]. Основная же опасность использования популярных систем заключается в расширении функционала веб-сайтов с помощью дополнительных модулей (плагинов) от сторонних разработчиков, не уделяющих проблеме безопасности должного внимания.

Исследована безопасность веб-сайтов, написанных на скриптовом языке PHP и обслуживаемых ЦИИР БГУИР. Был проведен поиск вредоносного ПО методами сигнатурного сканирования и эвристического анализа исходных файлов веб-сайтов. В результате на небольшом количестве сайтов (менее 50) было найдено более 200 образцов вредоносного программного обеспечения.

В абсолютном большинстве доступ был получен злоумышленниками через популярные системы управления содержимым веб-сайтов и их модули. Веб-сайты и системы, разработанные ЦИИР БГУИР не были скомпрометированы. При анализе исходных кодов были обнаружены образцы вредоносного ПО различного характера действия и разной степени шифрования.

С каждым днем вредоносный код (веб-шеллы, скрипты для рассылки спама и т.п.) становится более изощренным и сложным в обнаружении. Кроме обфускации идентификаторов и шифрования кода злоумышленники повсеместно начали использовать неявные вызовы функций посредством методов с callable аргументами, handler'ы и косвенные вызовы функций.

Вредоносных скриптов с линейной структурой и статическими переменными и функциями среди найденных образцов практически не было обнаружено. Исходный код стараются замаскировать и сделать как можно более изменчивым, «полиморфным» или наоборот, сделать максимально простым и похожим на обычный скрипт.

Вредоносное ПО размещается не только в новых файлах, а также и в системных файлах CMS, на которой функционирует сайт. В большинстве случаев это не влияет на работу сайтов, либо влияет незначительно, тем самым усложняя обнаружение атаки администраторами.

Зачастую, анализируя вредоносный скрипт, невозможно выделить фиксированный фрагмент, по которому однозначно можно было бы идентифицировать «вредонос». Очевидно, что подобный вредоносный код невозможно найти по простой базе сигнатур (антивирусной базе), которая используется в подавляющем большинстве веб-антивирусов и сканеров исходных кодов. Для эффективного поиска современных «вредоносов» необходимо использовать более сложные методики определения вирусных паттернов, а в некоторых случаях - эвристику.

В качестве минимизации риска сетевых атак на веб-сайты, использующие популярные системы управления содержимым (CMS) предлагается использовать следующие методы и средства:

- Использование сложных паролей к администраторским панелям любых сайтов и систем
- Ограничение доступа к панелям администрирования, ограничение попыток неудачного входа
- Запрещение регистрации пользователей при возможности
- Запрещение прав редактирования системных файлов CMS
- Запрещение возможности выполнения скриптов в тех папках, права на запись в которых невозможно выставить без сохранения функционирования CMS
- Регулярное создание резервных копий файлов и БД, на разных серверах и хранилищах
- Минимизация использования дополнительных модулей от сторонних разработчиков
- Регулярное обновление самих CMS и модулей к ним
- Осуществлять проверку файлов сканерами вредоносного кода (AI-Bolit, maldet, clamav и др.)

Все вышеописанные методы позволяют снизить риск сетевой атаки, но не гарантируют полную безопасность веб-сайтов. Максимально эффективным средством является использование систем обнаружения вторжений (IDS).

Для серверов ЦИИР БГУИР, на которых размещаются сайты, работающие на скриптовом языке PHP была выбрана система OSSEC - хостовая система обнаружения вторжений (Open Source Host-based Intrusion Detection System). Данная система обладает открытым исходным кодом, а значит ее использование безопасно. С ее помощью были решены задачи проверки контроля целостности файлов, логирования

различных действий на серверах, получения событий безопасности (анализ системных журналов) и оповещений об этих событиях.

В результате всех описанных методов и средств удалось существенно повысить безопасность сайтов, использующих популярные системы управления содержимым. Количество успешных сетевых атак злоумышленников значительно снизилось: на том же количестве сайтов за месяц наблюдения не было выявлено успешных атак. Проведенная работа подтверждает актуальность выбранного направления исследования информационной безопасности веб-узлов.

Список использованных источников:

17. W3Techs, Software Quality Management Consulting: [Электронный ресурс]. Режим доступа: http://w3techs.com/technologies/overview/content_management/all (Дата обращения: 11.03.2016).

18. Sucuri Security, Critical 0-day Remote Command Execution Vulnerability in Joomla [Электронный ресурс]. Режим доступа: https://blog.sucuri.net/2015/12/remote-command-execution-vulnerability-in-joomla.html (Дата обращения: 10.03.2016).

19. Wikipedia, OSSEC [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/OSSEC (Дата обращения: 10.03.2016).