**Полезные ссылки**

Про обфусцированный код

<https://www.linux.org.ru/forum/development/13231245>

<https://stackoverflow.com/questions/3328235/how-does-this-giant-regex-work>

<https://qna.habr.com/q/416937>

<https://www.linux.org.ru/forum/development/8158053>

База веб-шеллов

<https://github.com/alphaSeclab/awesome-webshell>

Про обфускацию

<https://habr.com/ru/articles/735812/>

<http://citforum.ru/security/articles/obfus/>

Деобфусцирование JS-кода в html

<https://habr.com/ru/articles/709304/>

Прочее

<https://habr.com/ru/companies/cleantalk/articles/351572/>

В первую очередь мы разбиваем исходный код на лексемы (минимальная языковая конструкция) и удаляем все ненужное[[1]](#footnote-1):

1. Символы пробелов.
2. Комментарий различных видов.
3. Не PHP код (вне тегов <?php ?>)

Далее мы рекурсивно упрощаем код, пока не останется «сложных конструкций»:

* Выполняем конкатенацию строк.
* Подстановку переменных в переменные.
* и прочее

Также в процессе упрощения кода мы следим за происхождением переменных и многим другим.

В итоге мы получаем чистый код, который можно анализировать. Очень важно, что код мы получаем не в виде строки, а в виде лексем. Таким образом, мы знаем, где находится лексема-строка с искомым текстом, а где лексема-функция.

Степени критичности:

1. Критические:
   * 1. eval
     2. include\* и require\*
        1. с плохим расширением файла
        2. несуществующих файлов
        3. подключение удаленных файлов
2. Опасные
   * 1. system
     2. passthru
     3. proc\_open
     4. exec
     5. include\* и require\*
        1. с оператором подавления ошибки
        2. с переменными зависящими от POST или GET.
3. Подозрительные
   * 1. base64\_encode
     2. str\_rot13
     3. syslog
4. И другие.

1. https://habr.com/ru/companies/cleantalk/articles/351572/ [↑](#footnote-ref-1)