Лабораторная работа №2. Обезвреживание двоичной бомбы. Срок выполнения: 11–25 марта 2013г.

1 Введение

Гнусный Доктор Зло заминировал наши виртуалки.

Двоичная бомба — это программа, которая состоит из последовательности фаз. Каждая фаза ожидает от вас, что вы напечатаете некоторую строку в stdin. Если строка будет правильной, то фаза обезврежена, и бомба передёт к следующей фазе. В противном случае бомба взорвётся, напечатав "БУМ!!!" и закончив работу. Бомба разряжена, если всё её фазы обезврежены.

Для нас тут бомб многовато, так что каждому студенту даётся по одной на разминирование. Ваша миссия — а вы не можете отказаться — обезвредить бомбу в срок. Удачи, и добро пожаловать в наш сапёрный отряд!

Шаг 1: Получите свою бомбу

Бомбу можно получить, набрав в браузере

http://s27000.vdi.mipt.ru:15213/

Появится форма запроса, которую нужно будет заполнить. Введите своё имя пользователя и пароль и нажмите "Submit". Сервер соберёт вашу бомбу и отправит вам её через браузер в tar-файле с названием bombk.tar, где k —это уникальный номер вашей бомбы.

Сохраните файл bombk.tar в (защищённую) директорию, в которой планируете работать. Потом наберите tar -xvf bombk.tar. Это создаст директорию ./bombk со следующими файлами:

- README: Определяет бомбу и её владельцев.
- bomb: Исполняемый файл с двоичной бомбой.
- bomb.c: Исходный код функции main бомбы и дружелюбное приветствие от Доктора Зло.
- writeup.{pdf,ps}: Описание лабораторной.

Если по какой-то причине вы запросили несколько бомб, это не проблема. Выберите одну, с которой будете работать, и удалите остальные.

Шаг 2: Обезвредьте свою бомбу

Ваша задача на этой лабораторной заключается в том, чтобы обезвредить свою бомбу.

Вы должны делать эту лабораторную на виртуальной машине. Есть даже слух, что Доктор Зло реально зол, и бомба всегда взрывается, если запустить её где-нибудь ещё. И ещё говорят, что бомба как-то защищена от подделок.

Для помощи в обезвреживании вы можете использовать различные инструменты. Пожалуйста, обратите внимание на подсказки в разделе Советы. Лучший способ — это использовать отладчик, чтобы пошагово пробраться через дизассемблированный двочиный файл.

Каждый раз, когда ваша бомба взрывается, она сообщает об этом на сервер, и от вашей итоговой оценки отнимается 1/2 балла (но не более 20 баллов всего). Так что бомба взрывается не без последствий. Будьте осторожны!

Первые четыре фазы стоят по 10 баллов. Пятая и шестая фазы немного сложнее, так что они стоят по 15 баллов. Так что максимальная оценка за эту лабораторную высчитывается из 70 баллов.

Хотя фазы постепенно и становятся сложнее, приобретаемый вами опыт должен компенсировать такой рост сложности. Учтите, однако, что последняя фаза может вызвать трудности даже у наиболее успевающих студентов, так что, пожалуйста, не откладывайте всё на последнюю минуту.

Бомба игнорирует пустые строки во входе. Если вы запустите бомбу с аргументом командной строки, например

```
linux> ./bomb sol.txt
```

то она будет читать строки из файла sol.txt, пока не достигнет его конца (EOF), а затем переключится на стандартный поток ввода (stdin). Минутная слабость заставила Доктора Зло добавить эту возможность, чтобы вам не пришлось заново набирать решения фаз, которые вы уже прошли.

Чтобы бомба случайно не взорвалась, вам нужно будет научиться выполнять ассемблерный код пошагово, а также выставлять точки останова. Вам также нужно будет научиться обследовать как регистры, так и содержимое памяти. Один из приятных сторонних эффектов этой лабораторной заключается в том, что вы научитесь пользоваться отладчиком. Это хорошее умение, и оно вам пригодится в будущем.

Как сдавать

Сдавать ничего не надо. Бомба сама сообщит преподавателю о ваших успехах. Вы можете оценить свой прогресс, посмотрев на общую таблицу по адресу:

http://s27000.vdi.mipt.ru:15213/scoreboard

Эта страница постоянно обновляется и показывает прогресс для каждой бомбы.

Советы (пожалуйста, прочтите!)

Существует несколько обезвредить бомбу. Вы можете довольно подробно исследовать её, не запуская программу, и понять в точности, что же она делает. Это полезный метод, но его не всегда легко привести в исполнение. Вы можете также запустить программу под отладчиком, посмотреть, что она делает на каждом шаге, и воспользоваться этой информацией для разминирования. Это, пожалуй, самый быстрый способ.

Мы хотели бы попросить, пожалуйста, не пытайтесь сломать бомбу перебором! Вы можете написать программу, которая попробует все возможные строки с целью найти нужную. Но это не очень хорошо, потому что:

- Вы теряете 1/2 балла (хотя и не более 20 баллов всего) каждый раз, когда вы не угадываете, и бомба взрывается.
- Каждый раз, когда вы не угадываете, на сервер с бомбой посылается сообщение. Если посылать их очень часто, такие сообщения могут быстро засорить сеть, и нам придётся забрать у вас доступ к машине.
- Мы не сказали вам, какой длины строки и сколько в них символов. Даже если вы сделаете (неверное) предположение о том, что они содержат не более 80 символов каждая и состоят только из букв, вам потребуется 26^{80} попыток для каждой фазы. Это будет работать очень долго, и вы не успеете получить ответ в срок.

Существует много инструментов, которые разработаны для того, чтобы понять, как работают программы, и в чём проблема, если они не работают. Вот список некоторых инструментов, которые могут вам пригодиться, а также несколько советов по их использованию.

- gdb

  Отладчик GNU (GNU debugger)—это средство для отладки с интерфейсом командной строки, доступное практически на любой платформе. С его помощью можно проследить за выполнением программы строка за строкой, осмотреть содержимое памяти и регистров, почитать как исходный, так и ассемблерный код (большую часть исходного кода бомбы мы вам не даём), поставить точки останова (breakpoints), установить наблюдение за интересующими вас областями памяти (memory watchpoints), а также писать скрипты.

  По адресу

  http://ubermipt.com/2sem

  доступен список команд gdb, который можно распечатать и использовать как справочник. Вот несколько советов по использованию gdb.

- Чтобы ваша бомба не взрывалась каждый раз, когда вы набираете неправильную строку, вам нужно научиться ставить точки останова.
- Для документации в режиме онлайн, наберите "help" в командной строке gdb, или наберите "man gdb" или "info gdb" в командной строке Unix. Некоторым также нравится запускать gdb с помощью режима gdb-mode редактора emacs или с помощью графического интерфейса ddd.

- objdump -t

  Эта команда напечатает таблицу символов бомбы. Таблица символов включает в себя все имена всех функций и глобальных переменных, имена всех функций, которые вызывает бомба, и их адреса. Что-то можно понять, просто посмотрев на имена функций!

- objdump -d

  Используйте эту команду для того, чтобы дизассемблировать весь код бомбы. Можно также просто посмотреть на отдельные функции. Чтение ассемблерного кода поможет вам понять, как работает бомба.

  Хотя команда objdump -d и даёт вам много информации, всю картину она не рисует. Вызовы некоторых системных функций могут отображаться в непонятной форме. Например, вызов sscanf может выглядеть так:

  8048c36: e8 99 fc ff ff call 80488d4 <_init+0x1a0>

  Чтобы определить, что это именно вызов sscanf, вам придётся дизассемблировать с помощью gdb.

- strings

  Эта программа покажет печатаемые строки в вашей бомбе.

Ищете какой-то конкретный инструмент? Как насчёт документации? Не забывайте, команды apropos, man и info — это ваши друзья. В частности, может оказаться полезным man ascii. info gas даст вам даже больше информации по ассемблеру GNU, чем вы захотите. Если всё совсем печально, обратитесь за помощью к преподавателю.

FAQ

— А будут какие-то плюшки за решение секретной фазы?
— Что ещё за секретная фаза? :-)