Lehrstuhl für Systemsicherheit
Prof. Dr. Thorsten Holz
Andre Pawlowski

Vorlesung Betriebssystemsicherheit
Wintersemester 2014/2015
Ruhr-Universität Bochum

3. Übung

24. November 2014
Abgabe der Hausaufgaben: per Moodle bis zum 8. Dezember 2014, 16:00 Uhr
Bei Fragen und Problemen können Sie sich per E-Mail/Moodle an die Betreuer wenden.

Bitte lösen Sie alle Linux-Aufgaben in der xubuntu-VM aus der letzten Übung.

Aufgabe 1 (Format Strings) (2 Punkte)

Ziel der Aufgabe ist es, mittels eines Format-String-Angriffs den Inhalt einer Speicherstelle auf einen bestimmten Wert zu setzen.

Schreiben Sie dafür einen Exploit, der das Programm `string` mit einem entsprechenden Kommandozeilenparameter aufruft, so dass die Erfolgsmeldung ausgegeben wird. Beschreiben Sie den Exploit außerdem in einem kurzen Kommentar im Quelltext.

Aufgabe 2 (Format Strings Code Execution) (2 Punkte)

Nutzen sie die Format-String-Schwachstelle aus der vorherigen Aufgabe aus, um Shellcode im System auszuführen. Sie können dazu beispielsweise eine Adresse im Global-Offset-Table (GOT) mittels eines entsprechenden Format-Strings überschreiben, um zum Shellcode zu springen.

Schreiben Sie einen Exploit als kleines Programm, das `string` aufruft und eine Shell erzeugt. Beschreiben Sie außerdem in einem kurzen Kommentar im Quelltext, wie der Exploit funktioniert. Um Probleme beim Korrigieren durch verschobene Adressen vorzubeugen, gehen wir davon aus das sich die Datei `string` im Ordner `/home/user/aufgabe1_2` befindet und auch der exploit aus diesem Ordner heraus ausgerufen wird.

Aufgabe 3 (Advanced Buffer Overflow) (2 Punkte)

Die Datei `exploitme.c` enthält eine Buffer-Overflow-Schwachstelle ähnlich wie auf dem letzten Übungsblatt. Allerdings variiert der Stackpointer bei jeder Ausführung zu einem gewissen Grad. Schreiben Sie einen Exploit, der eine Remote-Shell auf einem Netzwerkport Ihrer Wahl erzeugt, *ohne* einen NOP-Slide zu verwenden. Ansonsten ist jede Technik erlaubt.

Schreiben Sie einen Exploit als kleines Programm, das `exploitme` aufruft und über die Schwachstelle eine Shell erzeugt. Beschreiben Sie außerdem in einem kurzen Kommentar im Quelltext, wie der Exploit funktioniert. Um Probleme beim Korrigieren durch verschobene Adressen vorzubeugen, gehen wir davon aus das sich die Datei `vuln` im Ordner `/home/user/aufgabe3` befindet und auch der exploit aus diesem Ordner heraus ausgerufen wird.

Aufgabe 4 (Hidden Format String Vulnerable) (Bonusaufgabe, 0 Punkte)

Diese Aufgabe ist etwas schwieriger und deshalb unbenotet. Bei der Aufgabe handelt es sich um eine Aufgabe aus einem CTF und hat mehrere Tricks eingebaut. Um die Aufgabe auf den Format String zu reduzieren, bekommen Sie hier vorab die nötigen Schritte. Achten Sie darauf, dass die Datei `salt.txt` im selben Verzeichnis liegt wie das Programm selber. Wenn Sie das Programm ausführen, werden Sie als nächstes nach einem geheimen Slogan gefragt. Hier lautet die Lösung "gold>silverb". Die eigentliche Format String Lücke ist zu finden, in dem man eine neue Mine anlegt ("1) Add mine") und sich den Profit dieser Mine ("4) Show profit") ausgibt. Welcher dieser Werte genau den Format String auslöst, müssen Sie selber herausfinden.

HINWEIS: Der Format String wird durch den Aufruf der Funktion `fputs()` an der Adresse `0x8049184` ausgelöst. Die `fputs()` Funktion ist normalerweise nicht gegen einen Format String verwundbar, in dieser Aufgabe aber schon. Die Funktionsnamen werden durch naives Verhalten in den Analyseprogrammen verfälscht. Hier die Liste der verfälschten Funktionsnamen:

- `printf()` wird verfälscht zu `fputs()`
- `strncmp()` wird verfälscht zu `strcmp()`
- `system()` wird verfälscht zu `printf()`

Deshalb ist die oben angegebene Adresse für `fputs()` in Wirklichkeit ein Aufruf von `printf()`. Am einfachsten ist es diese Lücke auszunutzen, wenn Sie die lokal geladene `system()` Funktion mit der Adresse `0x80485e6` benutzen.

Falls Sie einen groben Überblick über das komplette Programm haben wollen, können Sie dies am einfachsten mit dem Programm IDA:

`https://www.hex-rays.com/products/ida/support/download_freeware.shtml`

erreichen.