Advanced Unix System Administration

Lecture 19
April 11, 2007

Steven Luo
<sluo+decal@OCF.Berkeley.EDU>

After a Compromise

- Don't panic!
  - Take (at least) a few moments to assess the situation clearly
  - You want to get a quick idea of the potential impact of the compromise, so you have an idea of what needs to be done to contain the damage
  - Resist the temptation to pull the plug right away! You'll likely lose valuable information about the attack

After a Compromise

- Damage containment
  - What's the known scope of the compromise?
    - Isolate the affected machines and/or services as soon as practical
    - Restrict access to data, logs, etc. from these machines if at all possible
  - What else could the attacker have done?
    - Inspect other machines and services that could have been affected carefully
    - It may be worth it to go into a “lockdown” mode until inspection is completed

After a Compromise

- Attack analysis
  - In general, the focus of the investigation should be how the attacker got in, not how to clean up
    - Thoroughly cleaning up a root compromise by a good attacker is difficult and dangerous, so it's safest to reinstall affected machines from trusted media
    - You want to know how the attacker got in so you can tighten up your security in future – it's not going to do you any good if you reinstall the same way and get r00ted again

After a Compromise

- Attack analysis con't
  - You can't trust anything coming from a potentially-compromised system
    - It's common for an attacker to replace most of the useful system tools (ps, ls, etc.) with compromised versions
    - If at all possible, get a copy of the tools from read-only media and work from those
    - In the case of a kernel-level rootkit, it may not be possible to recover any useful information at all from the running system

After a Compromise

- Before shutting down the system
  - Files and directories
    - Check timestamps, look for recently modified files and directories, use your checksum database if you have one
      - The stat(1) command is useful for showing timestamps
      - Timestamps may not be accurate, but depending on how thorough the attacker was, you may be able to build a timeline of the attack
    - Files that shouldn't be there, files with weird names (“...”, “..[space]”, '\n', etc.)
      - These are favorite places for attackers to hide information

After a Compromise

- Before shutting down the system con't
  - Avoid writing to the disks of the compromised system!
    - You may inadvertently overwrite something useful, or trigger a logic bomb
  - Running processes
    - Remember that argv[0] is not a reliable way of determining what a process is
    - Poking around in /proc and/or applying strace/truss can give you a fair lot of information – though be careful

After a Compromise

- Before shutting down the system con't
  - Network services
    - See if you can spot anything that shouldn't be running
    - It's not a good idea to connect to rogue network services
  - Logs (you did keep them, right?)
    - System, process and accounting logs can be invaluable in establishing a timeline
    - Beware of tampering – fake log entries, incorrect timestamps, gaps in logs, etc.
    - Logs kept off the affected machines help greatly

After a Compromise

- Shutting down the system
  - Do an unclean shutdown (pull the power) – you'll lose less evidence that way
  - Boot the system from known-good media
  - Repeat the filesystem checks, log reading, and such
    - This will tell you whether or not your previous efforts were tampered with
    - Keep the filesystems mounted read-only!
  - Consider taking filesystem images for further analysis

After a Compromise

- Preventing it from happening again
  - Assemble your evidence and try to construct a timeline for the attack
    - You want to know when the attacker did what, and why it worked (or didn't work)
  - From the timeline, identify improvements your security
    - This might include your procedures and only partially-related configuration changes!
  - Implement that better security