Lecture 24: The Dictionary Attack and the Rainbow-Table Attack on Password Protected Systems

Lecture Notes on "Computer and Network Security"

by Avi Kak (kak@purdue.edu)

April 18, 2017

2:30pm

Goals:

- The Dictionary Attack
- Thwarting a dictionary attack with log scanning
- Cracking passwords with direct table lookup
- Cracking passwords with hash chains
- Cracking password with rainbow tables
- Password hashing schemes

CONTENTS

24.1: THE DICTIONARY ATTACK

- Scanning blocks of IP addresses for vulnerabilities at the ports that are open is in many cases the starting point for breaking into a network.

- If you are not behind a firewall, it is easy to see such ongoing scans. All you have to do is to look at the access or the authorization logs of the services offered by a host in your network. **You will notice that the machines in your network are being constantly scanned for open ports and possible vulnerabilities at those ports.**

- In this lecture I will focus on how people try to break into port 22 that is used for the SSH service. This is a **critical** service since its use goes way beyond just remote login for terminal sessions. It is also used for secure pickup of email from a mail-drop machine and a variety of other applications.

- The most commonly used ploy to break into port 22 is to mount what is referred as a **dictionary attack** on the port. In a dictionary attack, the bad guys try a large number of commonly used names as possible account names on the target machine and, should they succeed in stumbling into a name for which there is actually an account on the target machine, they then proceed to try a large number of commonly used passwords for that account. [An attack closely related to the dictionary attack is known as the **brute-force attack** in which a hostile agent systematically tries **all** possibilities for usernames and passwords. Since the size of the search space in a brute-force attack increases exponentially with the lengths of the usernames and passwords used in the attack, it is not generally feasible to mount such attacks through the internet.]

- If you are logged into a Ubuntu machine, you can see these attempts on an ongoing basis by running the following command line in a separate window

```
tail -f /var/log/auth.log  |  sed G
```

- I will now show just a **two minute segment** of this log produced not too long ago on the host `moonshine.ecn.purdue.edu`. To make it easier to see the usernames being tried by the attacker, I have made a manual entry in a separate line for just the username that the attacker tries in the next break-in attempt. Note that the third line shown for each break-in attempt is truncated because it is much too long. Nonetheless, you can see all of the relevant information in what is displayed. This scan was mounted from the IP address `61.163.228.117`. If you enter this IP address in the query window of `http://www.ip2location.com/` or `http://geoiptool.com`, you will see that the attacker is logged into a network that belongs to the The Postal Information Technology Office in the city of Henan in China.

```
username tried: staff

Apr 10 13:59:59 moonshine sshd[32057]: Invalid user staff from 61.163.228.117
Apr 10 13:59:59 moonshine sshd[32057]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 13:59:59 moonshine sshd[32057]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:01 moonshine sshd[32057]: Failed password for invalid user staff from 61.163.228.117 port 40805 ssh2


username tried: sales

Apr 10 14:00:08 moonshine sshd[32059]: Invalid user sales from 61.163.228.117
Apr 10 14:00:08 moonshine sshd[32059]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:08 moonshine sshd[32059]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:10 moonshine sshd[32059]: Failed password for invalid user sales from 61.163.228.117 port 41066 ssh2


username tried: recruit

Apr 10 14:00:17 moonshine sshd[32061]: Invalid user recruit from 61.163.228.117
Apr 10 14:00:17 moonshine sshd[32061]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:17 moonshine sshd[32061]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:19 moonshine sshd[32061]: Failed password for invalid user recruit from 61.163.228.117 port 41303 ssh2


username tried: alias

Apr 10 14:00:26 moonshine sshd[32063]: Invalid user alias from 61.163.228.117
Apr 10 14:00:26 moonshine sshd[32063]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:26 moonshine sshd[32063]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:29 moonshine sshd[32063]: Failed password for invalid user alias from 61.163.228.117 port 41539 ssh2


username tried: office

Apr 10 14:00:36 moonshine sshd[32065]: Invalid user office from 61.163.228.117
Apr 10 14:00:36 moonshine sshd[32065]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:36 moonshine sshd[32065]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:38 moonshine sshd[32065]: Failed password for invalid user office from 61.163.228.117 port 41783 ssh2


username tried: samba

Apr 10 14:00:46 moonshine sshd[32067]: Invalid user samba from 61.163.228.117
Apr 10 14:00:46 moonshine sshd[32067]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:46 moonshine sshd[32067]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:47 moonshine sshd[32067]: Failed password for invalid user samba from 61.163.228.117 port 42027 ssh2


username tried: tomcat

Apr 10 14:00:55 moonshine sshd[32069]: Invalid user tomcat from 61.163.228.117
Apr 10 14:00:55 moonshine sshd[32069]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 14:00:55 moonshine sshd[32069]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rho
Apr 10 14:00:57 moonshine sshd[32069]: Failed password for invalid user tomcat from 61.163.228.117 port 42247 ssh2


username tried: webadmin
```