

SSH PROBES: THE PRICE OF UNIX MAINSTREAMING?

Kaido Kikkas

Centre for Educational Technology, Tallinn University
Narva Road 25, 10120 Tallinn, Estonia
kaido.kikkas@kakupesa.net

NB! THIS IS THE PREPRINT VERSION

ABSTRACT

Security has gradually risen to be one of the key fields in today's Internet. Easily-operated personal computers and fast broadband connections have multiplied the user base - at the cost of the average technical knowledge level. This has resulted in epidemic spread of malware, viruses and worms as well as 'computer hijacking' over the net, giving offenders control over a large number of PCs. Until recently, the attention was mostly focused on Windows systems, but the gradual mainstreaming of Unix-derived systems like Linux have made them easier targets. The paper provides a short analysis of the situation, focusing on one of the simpler type of network attacks: SSH brute force probes. A one-year survey of SSH attempts on a server is used as a real-life example.

KEYWORDS

SSH, Unix, Linux, script attacks, passwords, security

1. INTRODUCTION

During the last decades, the main target of attacks has been Microsoft Windows. While the latest XP has introduced a number of features to improve security, it remains haunted by all kinds of malware, viruses and worms (see Leyden 2005ab and Secunia.com). By contrast, Unix-based systems like Linux and BSD have been widely regarded as more secure. While this assumption is not baseless (a good comparison is Petreley 2004), the user base has increasingly become the determining factor. Labelled as 'geek system' and 'hacker only', many of these systems frightened away all but the most knowledgeable, resulting in small but highly motivated and professional user base. These days, especially Linux is backed by large companies (IBM, Red Hat, Sun, HP and others) and marketed as an alternative to dominating MS Windows. While this tendency has many good aspects (less monopolism, more choice, lower costs resulting from stronger competition etc), there is an alarming side – not any more the system for “the elite”, the average level of users' technical skills and knowledge is bound to fall quite notably.

Perhaps one of the most flammable combinations increasingly threatening the integrity of the very Internet is the wide spread of broadband-connected personal computers with inadequate security measures. While the benefits of everyone's Internet can hardly be denied, the lax security of personal computers has led to *en masse* compromises and hijacks, where hundreds of computers at homes of unsuspecting owners can be used in coordinated manner to attack some major target in Internet. Also, these *zombies* (compromised machines) are responsible for a large share of unsolicited e-mail (*spam*) and other negative phenomena. Until recently, it was easy to blame it all on the „Big Bad Microsoft“ - after all, Windows does have a long history of security problems. However, as Linux (and other „alternative“ systems) moves towards mainstream, it will be targeted more as well. And even if it comes with adequate security mechanisms, no one can protect users from their own incompetence.

2. SSH

SSH (*Secure SHell*) is a major protocol used in today's Internet communication. It has largely replaced the earlier standard for remote access – *telnet*. Telnet has open, non-encrypted traffic that is possible to intercept and eavesdrop, while SSH offers encryption which makes getting information out of intercepted messages much more difficult. The same technology is also widely used in file transfer applications (*sftp* and *scp* are increasingly preferred to the older FTP standard).

SSH-based probing attacks are one of the simplest attack forms, basically being a sort of „door-testing“ with a number of widely-used usernames (adam, admin, alan, alex...) and simple or no password. As running SSH services are more common in Unix systems (Windows XP does not include an SSH server by default, although it can be installed separately), this is one of the threat categories which does not have Windows as its main target.

Although non-complex, the probes have proven effective in many cases. Many Linux distributions tend to install and run SSH servers by default, which is often not so evident for a more casual user. And if these features combine with loose attention towards passwords (which regrettably is a trait which characterizes a large majority of Windows-trained users – and if they eventually switch to other systems, they tend to keep their habits), we get quite a dangerous combination.

What makes things worse – unlike many sophisticated attacks, SSH attack scripts are simple and easy to use - a good example is BruteSSH (see FrSIRT and LinuxQuestions) with its many variants - which makes them a perfect tool for *script kiddies* (the term is used for mostly junior offenders who lack deep technical understanding about the tools they use; see Wikipedia). Many of the tools are of the 'fire and forget' type – a kiddie can launch a program late at night, go to bed and retrieve the list of matched usernames and passwords in the morning. Meanwhile the program has scanned a big portion of Internet and tried a predetermined (dictionary-based) set of usernames and passwords on each SSH-running computer it encountered. Compromised systems may then be attacked directly (defacement of web pages, erasure of content etc), but more likely are they used to carry out a large variety of illicit operation from password sniffing to spam propagation to DDOS (*distributed denial of service* attack – using a coordinated network of machines to attack a single target, effectively killing it by excessive traffic or sometimes opening up a security hole by the overload).

3. THE SURVEY

The server used in the study is a home-based server. Its owner works at three different universities and uses the server as the sole channel of study materials (which greatly enhances the otherwise relatively limited visitor base) as well as send/receive e-mail and act as a gateway for home network (which includes some neighbours). The network connection used was 2Mbit/640kbit ADSL for most of the survey time, switching to 1Mbit ADSL during the last month of the study. During the study, the machine ran White Box Linux 3 operating system (a free clone of Red Hat Enterprise Linux 3; later upgraded to CentOS 4, another clone) and is reasonably well protected, which has resulted in no security incidents since its launch in early 2002.

SSH attempts on the server were monitored during one year – starting from the first occasion on July 17, 2004 and ending on July 17, 2005. During the period, the number of attempts increased drastically – while at the beginning it was possible to send private e-mail notices to owners of the offending computers (as identified by *whois* utility), the practice had to be discontinued due to increasing volumes. All in all, the total number of SSH attempts during the year was 32 370.

The information was collected by using the LogWatch utility, which recorded both the offending IP addresses and host names. These can be counterfeited (*spoofed*), but in most cases use of complementary utilities (*ping*, *whois*, *traceroute* etc) pointed to the initial directions, i.e. a majority of attempts did not bother to cover their tracks (also the notifications that were sent out during the early stages were mostly replied with 'thank you for pointing out the compromise' messages). This can largely be explained by the use of *zombies* or hijacked computers owned by someone else than the real attacker.

4. RESULTS

The first SSH attacks were registered on July 17, 2004. For quite a long time, the number of attempts was quite limited, using only the most obvious usernames. A noteworthy mention is that among these early attempts, the username *test* is prevalent – both with and without password. While the number of attempts has increased over time, *test* is still included in most attacks. Anecdotal evidence from Internet seems to prove that archetypal temporary account names are favoured due to it being treated with less care than regular ones by many administrators. Another similar username is *guest* which apparently has also been created for temporary users, insufficiently monitored and possibly later forgotten to remove (see also CounterPane.com).

By autumn the number of attempts increased, going over 100 in October with the appearance of scripts which tried a large number of common usernames. Total numbers were still quite low, but from the day one the attempts were rather evenly spread over the year, the longest break between the attempts being 4 days. A typical day had 2-3 different attackers with varying number of attempts. By the end of the year, the top result was 400 attempts in a day, but the absolute record came near the end of the survey as the July 13, 2005 gave as many as 2465 attempts.

The geographical distribution is also quite interesting. Many Internet attacks are widely associated with Far East countries (see Tran 2005) and the survey showed this as well – 101 attacks had come from Korea (2nd), 71 from China (3rd), 39 from Taiwan (4th) and 18 from Japan (5th). The top position (in both attack and attempt numbers) was still held by the United States – 105 attacks with 11 300 attempts. In Europe, most attacks were coming from larger countries - the UK, Germany, France and Italy. Estonia had only 1 attempt and its northern neighbour with good renown in hi-tech – Finland – had none.

Although the identity of the computers used for attacks cannot be fully verified, use of additional utilities and mail responses allow to assume that most of the IP-s shown were genuine. Some of the more interesting places included Royal Institute of Technology in Sweden, *Société Européenne des Satellites* in Luxembourg, *Leibniz-Rechenzentrum* and *Hochschule für Kunst Bremen* in Germany, *Centre de Calcul de l'Université Bourgogne* in France, Zagreb University in Croatia, University of Colorado and National Association for Child Care in the US, University of Saskatchewan in Canada, Ministry of Education Computer Center in Taiwan, Kumamoto Gakuen University in Japan... The list includes many public facilities with presumably ample IT competence – this might hint how widespread the problem really is.

Also a substantial share of attacks resulted from the Internet service providers riding the global broadband wave – including Bellsouth.net in the US, Hanaro in Korea, *Cable i Televisio y Catalunya* in Spain/Catalunya. It is possible that providers do not pay enough attention to the security of their clients.

Finally, a small point to show the simple nature of SSH scripts – a large majority of them only try English-based usernames, making it less likely to score a hit in other countries with different cultures. Although some attempts included other languages (e.g. Finnish and Japanese), real usernames of the server were never matched during the year.

5. PROTECTIVE MEASURES AGAINST SSH PROBE ATTACKS

The first and foremost measure to avoid SSH script attacks is to educate users and set up a strict policy on passwords. These should be long enough and contain various letters, numbers and punctuation (see US-CERT). The user training is a complex issue which should be addressed much more seriously than today – it takes a joint effort of computer manufacturers and sellers, software distributors, Internet service providers and most of all, users themselves, who should be directed more towards online support communities (see also Kikkas 2005). Also all levels of education that provide IT training should focus on these problems.

Coming to more specific measures, many Unix-type systems warn the user if their chosen password is too simple – these recommendations must be followed. In Unix, most of the SSH tuning goes via a file called *sshd_config* (which is human-readable plain text file and has quite self-explaining settings) – the *PermitEmptyPasswords* should be set to *No*.

Administrator accounts (*root* in most Unix-derived systems) must not be remotely accessible over SSH. The option *PermitRootLogin* in the *sshd_config* file should be set to *No*. Another good idea, especially with smaller userbase, is to list all users entitled to login under the *AllowUsers* option in the *sshd_config*. There are also settings which limit the number of clients and the allowed time to connect. Finally, it makes sense to only use the newer version 2 of SSH, setting the *Protocol* to 2 (opposed to default 2,1). See OpenSSH Manual for more information.

There is an option to disable passwords altogether and use host- or key-based authentication. This is a viable way if only few computers can legally access the server via SSH, or the necessary key can be carried along (e.g. on a memory stick). In case of larger system with a lot of distributed users this way can be unsuitable. In this case, use of a blacklisting script like DenyHosts is recommended – it will register failed logins and exceeding a certain limit, list the offending addresses in the *hosts.deny* file to be blocked in the future (at the end of the survey, DenyHosts was applied to the server, resulting in rapid decline of SSH attempts).

An additional measure which proved itself well during the survey is to prefer nontrivial usernames. While an Italian youngster named Giovanni may think it cool to get his username as *johnny*, it is not a good idea as the most common English names tend to be the most likely targets.

6. CONCLUSION

While SSH brute force attempts with their simple tools represent the lower end of the spectrum of network attacks, they are dangerous enough. Due to Unix-derived systems moving into mainstream use, they are increasingly being used by non-technical people, which creates new risks. Distributors of Unix-like systems should consider the threat and provide default configurations with reasonably safe options. User training should be given much more importance, possibly being a joint task for system distributors, computer sellers and user communities on Internet.

REFERENCES

- Leyden, J. (2005) Zombie bots fuel spyware boom. *The Register*, July 11, 2005.
http://www.theregister.co.uk/2005/07/11/malware_report_mcafee/
- Leyden, J. (2005) Malware maelstrom menaces UK. *The Register*, July 18, 2005
http://www.theregister.co.uk/2005/07/18/malware_blitz/
- Secunia.com. <http://secunia.com/>
- Petreley, N. (2004) Security Report: Windows vs Linux. *The Register*, October 22, 2004.
http://www.theregister.co.uk/security/security_report_windows_vs_linux/#singleuser
- CounterPane Internet Security. <http://www.counterpane.com/alert-cis20040910-1.html>
- LinuxQuestions: SSH login attempts thread.
<http://www.linuxquestions.org/questions/showthread.php?s=&threadid=215431>
- FrSIRT: SSH Brute Force Exploit. <http://www.frstirt.com/exploits/08202004.brutessh2.c.php>
- Wikipedia: Script kiddie. http://en.wikipedia.org/wiki/Script_kiddie
- Tran, M. (2005). Government networks targeted by Asian hackers. *The Guardian*, June 16 2005
<http://www.guardian.co.uk/business/story/0,3604,1508097,00.html>
- US-CERT: Choosing and Protecting Passwords. <http://www.us-cert.gov/cas/tips/ST04-002.html>
- Kikkas, K. (2005). PC + Windows + Broadband + Ignorance = Disaster. <http://www.kakupesa.net/kakk/rant/>
- OpenSSH Manual. <http://www.openssh.org/manual.html>
- DenyHosts. <http://denyhosts.sourceforge.net>