

Hack Away!

Tools and technologies for security assessments and penetration testing

Kevin Spett

Internet Security Systems



 **INTERNET | SECURITY | SYSTEMS®**



- I am a consultant with Internet Security Systems.
- Typical services include external penetration testing, vulnerability assessments, source code reviews and remediation of any discovered problems.
- Customers include a variety of well-known financial, healthcare, retail, energy, municipal and other organizations.
- Previously worked in R&D at a web application security assessment software company.
- Spoken at several large technical computer security conferences.
- Discovered major vulnerabilities and written several penetration testing whitepapers.

Goals of this presentation



- **The goal of this presentation is to educate people about:**
 - What hacking is
 - Who hackers are
 - Why penetration testing is valuable
 - Tools and techniques that are useful



- **In the context of computer security, hacking is:**
 - The process of attempting to gain unauthorized access to a computer system.
 - Generally a criminal offense, although laws are notoriously vague and open to interpretation.
 - Sometimes as simple as guessing a password.
 - Sometimes as complicated as spending days in front of a debugger trying to decipher arcane assembly routines and undocumented operating system internals.

Who hackers are



- Hackers are usually young males who are intelligent and curious with lots of time on their hands.
- There are no good estimate of how many “hackers” are out there, but the ubiquity of the Internet and explosion of publicly available books and resources on the subject guarantee that their ranks have increased exponentially over the last ten years.
- Only a fraction of criminal hackers are arrested and convicted. It takes a lot of time, money and mistakes on their part to track them down. Is it really worth trying to get the Russian police to arrest a bored 17 year old and extradite him to the US to be prosecuted?
- Hackers can be divided into three groups based on skill level. As skill level increases, the amount of people in the group decreases, like a pyramid.



■ **Novice**

- Derided as “script kiddies”. They exclusively use very simple tools that others create. Most of the time they do not actually understand how these tools work or what they are doing. Often then will break into whatever they are able to right away indiscriminately. Most self-described “hackers” fit into this category. A good administrator will be able to protect against these attackers easily.



■ **Intermediate**

- These are people who have spent time trying to learn how programs and protocols actually work. They still mostly use tools written by other people, but are able to understand and wield more sophisticated and powerful ones. They select specific targets and persistently research and pursue them. It can be a challenge to defend against knowledgeable attackers who are willing to make this kind of effort.



■ **Advanced**

- These are the people that write their own tools and research their own vulnerabilities. They have very deep comprehension of computer systems and the software that they use. When they decide what their goal is, they will thoroughly and patiently work towards it. Most of the time, they will be successful. It is extremely difficult to protect a system from this group. Thankfully, their numbers are limited.



Who hackers are

- **Having a firewall, IPS and fairly recent patches and on exposed systems will stop script kiddie attacks.**
- **The very best hackers will determine what third party hardware and software you are running and can reverse engineer it to find new vulnerabilities. There is very little that can be done to stop these attackers.**
- **The major challenge for security administrators is to protect against the mid-level hackers who are skilled enough to know how to find and exploit vulnerabilities on a network using existing tools. This presentation focuses on methods and tools that are used by this group.**

Why penetration testing is valuable



- **Most IT departments are already busy. Asking them to go through every single server and network device, double check their configuration, do QA and testing on all available patches and fix any problems is very expensive in time, money and sanity.**
- **Computer security is easy for attackers and difficult for defenders. A system is only as secure as its weakest link. The defenders must secure every single element of the network, which is a lot of work. An attacker just needs to find one single problem, which is comparatively easy.**
- **Performing regular penetration testing is a good way to identify the most serious problems quickly and gauge risk without having to resort to an internal top-to-bottom audit of every component.**

Tools and techniques



- **There is an enormous amount of security tools available for free with complete source code. New ones are released all of the time.**
- **Commercial tools generally offer superior usability, completeness, documentation and support. However, an experienced and skillful attacker with a set of simple tools is far more capable than a novice with a \$50,000 push-button scanner. There are very few commercial products that have features that no free tools do.**
- **This presentation uses a variety of well-known programs that are used by both hackers and security professionals. They cover many different fields: network enumeration, password brute forcing, buffer overflow exploits, host configuration enumeration and cryptographic attacks.**

Tools and techniques - scanrand



```
Terminal — ssh — 79x25
$ scanrand
Destination required.
scanrand 1.10: Stateless TCP Scanner w/ Inverse SYN Cookies(HMAC-SHA1/32 in SEQ
)
Component of: Paketto Keiretsu 1.10; Dan Kaminsky (dan@doxpara.com)
Example: scanrand -b10M 10.0.1.1-254:80,20-25,139
Def. Ports: Use [quick/squick/known/all] instead of explicitly naming ports
Options: -S/-L: Only send requests / Only listen for responses
-e/-E: Show negative responses / Only show negative responses
-t [timeout]: Wait n full seconds for the last response (10s)
-b[bandwidth]: Limit bandwidth consumption to n b/k/m/g bytes(0)
(0 supresses timeouts or maximizes bw utilization
)
-N/-NN : Enable name resolution (Prefer Source/Dest)
-v : Mark packets being sent, as well as received
-vv : Output full packet traces to stderr
Addressing: -d [device]: Send requests from this L2 hardware device
-i [source]: Send requests from this L3 IP address
-p [ port]: Send requests from this L4 TCP Port
-s [ seed]: Use prespecified seed for scan verification
-f [ file]: Read list of targets from file
Experiments: -l [ttl-ttl]: Statelessly TCP Traceroute
-D : Distco (Distance Discover) via forced RSTs
-c : Try checking Inverse SYN Cookie on Traceroute
Notes: Use Control-C to exit before scanrand times out.
```

Tools and techniques - scanrand



- **scanrand is port scanner, which is used to scan network space for available services. This is usually one of the very first things an attacker will do when trying to compromise a network.**
- **It does not use the normal networking APIs and operating system TCP/IP stack, which are not optimized for sending out millions of requests to servers that may not exist. It uses its own state table and raw sockets.**
- **nmap, the most popular network scanner, has many additional features that scanrand does not. However, it cannot beat scanrand in terms of sheer performance for simple network scanning. Also, nmap has already been covered heavily in other presentations over the years.**



Tools and techniques - hydra

```
Terminal — ssh — 79x25
$ hydra
Hydra v5.0 [http://www.thc.org] (c) 2005 by van Hauser / THC <vh@thc.org>

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e ns]
[-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-f] [-s PORT] [-S] [-vV]
server service [OPT]

Options:
-R      restore a previous aborted/crashed session
-S      connect via SSL
-s PORT if the service is on a different default port, define it here
-l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
-p PASS or -P FILE try password PASS, or load several passwords from FILE
-e ns   additional checks, "n" for null password, "s" try login as pass
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE server list for parallel attacks, one entry per line
-o FILE write found login/password pairs to FILE instead of stdout
-f      exit after the first found login/password pair (per host if -M)
-t TASKS run TASKS number of connects in parallel (default: 16)
-w TIME defines the max wait time in seconds for responses (default: 30)
-v / -V verbose mode / show login+pass combination for each attempt
server  the target server (use either this OR the -M option)
service the service to crack. Supported protocols: [telnet ftp pop3 imap sm
b smbnt http-head http-get https-head https-get http-proxy cisco cisco-enable v
nc ldap2 ldap3 mssql mysql oracle-listener postgres nntp socks5 rexec rlogin pc
```



Tools and techniques - hydra

- **hydra is a password brute forcing tool. Thanks to its modular design, it supports over 30 different protocols. Whenever possible, it will make many simultaneous authentication attempts in parallel. This speeds up the password guessing process dramatically.**
- **Passwords are by far the most commonly used authentication system. As users are quick to admit, people do not like using complicated passwords. Most penetration tests uncover weak passwords.**
- **Guessing a password is still one of the simplest and most effective hacking techniques. The only thing that seperates your email from the entire Internet is a password.**

Tools and techniques - metasploit



```
Terminal — ssh — 79x25
$ ./msfconsole

+ -- --=[ msfconsole v2.5 [105 exploits - 74 payloads]

msf > show exploits

Metasploit Framework Loaded Exploits
=====

3com_3cdaemon_ftp_overflow      3Com 3CDaemon FTP Server Overflow
Credits                        Metasploit Framework Credits
afp_loginext                   AppleFileServer LoginExt PathName Overflow
aim_goaway                     AOL Instant Messenger goaway Overflow
altn_webadmin                  Alt-N WebAdmin USER Buffer Overflow
apache_chunked_win32           Apache Win32 Chunked Encoding
arkeia_agent_access            Arkeia Backup Client Remote Access
```

Tools and techniques - metasploit



- **metasploit is a suite of programs for vulnerability exploitation that has many features. It contains a very easy-to-use console for testing and exploiting remote server vulnerabilities. Dozens of working exploits and payloads are included.**
- **A modular design and APIs allow more advanced users to develop re-usable components for buffer overflow exploitation. Other parts of metasploit aid in new vulnerabilities research.**
- **Metasploit makes exploiting buffer overflow vulnerabilities very easy. Users do not need to know anything about what the vulnerabilities are or how the exploits work. You could probably train a monkey to use it.**

Tools and techniques - pwdump2



```
Terminal — ssh — 79x25
$ cat pwdump2.c
/*****
*
* File:      pwdump2.c
*
* Purpose:  dump the password hashes from the NT SAM.
*
* Date:     Sun Jun 07 12:46:59 1998
*
* Copyright (c) 1998, 2000 Todd A. Sabin, all rights reserved
*
* This program is free software; you can redistribute it and/or
* modify it under the terms of the GNU General Public License
* as published by the Free Software Foundation; either version 2
* of the License, or (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program; if not, write to the Free Software
* Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.
*
*****/
```

Tools and techniques - pwdump2



- **pwdump2 extracts user names and password hashes from NT's SAM database.**
- **A password hash is the result of a mathematical operation on a plaintext password. The OS stores the hash value instead of the actual password. When a user attempts to authenticate, the entered password is hashed, and the result is compared to the stored value. If they match, the user is authenticated.**
- **Unless they are very carefully configured, NT servers may use a weak password hashing procedure that can allow the original password to be recovered from the hash quickly.**

Tools and techniques - rcrack



```
Terminal — ssh — 79x25
$ rcrack
RainbowCrack 1.2 - Making a Faster Cryptanalytic Time-Memory Trade-Off
by Zhu Shuanglei <shuanglei@hotmail.com>
http://www.antsight.com/zsl/rainbowcrack/

usage: rcrack rainbow_table_pathname -h hash
       rcrack rainbow_table_pathname -l hash_list_file
       rcrack rainbow_table_pathname -f pwddump_file
rainbow_table_pathname: pathname of the rainbow table(s), wildchar(*, ?) supported
-h hash:                use raw hash as input
-l hash_list_file:      use hash list file as input, each hash in a line
-f pwddump_file:        use pwddump file as input, this will handle lanmanager hash only

example: rcrack *.rt -h 5d41402abc4b2a76b9719d911017c592
         rcrack *.rt -l hash.txt
         rcrack *.rt -f hash.txt

$ █
```

Tools and techniques - rcrack



- **rcrack is a tool that implements a time/memory trade-off attack against weak hashes.**
- **Strong password hashes go through a procedure called “salting” which places extra steps in the mathematical function to make it more complicated and difficult to break. This ensures that the attacker must brute force every possible password for each hash each time instead of just preparing a list of every single possible hash and doing a look up.**
- **Many Windows systems do not perform salting which makes it possible to generate a directory of all possible hashes and simply do a lookup to discover the plaintext.**

Live demonstration



- **These five tools and a realistic lab environment are used in a demonstration to show how a penetration test might proceed.**
- **The ultimate goal for the attacker is to gain access to confidential corporate email.**
- **While the email server does not immediately appear to be vulnerable, this does not mean that it is impossible to compromise it.**
- **After several seemingly unrelated systems are penetrated, a way to access the mailserver emerges.**



Feel free to ask questions about any of the material covered in this presentation or related topics.

kspett@iss.net