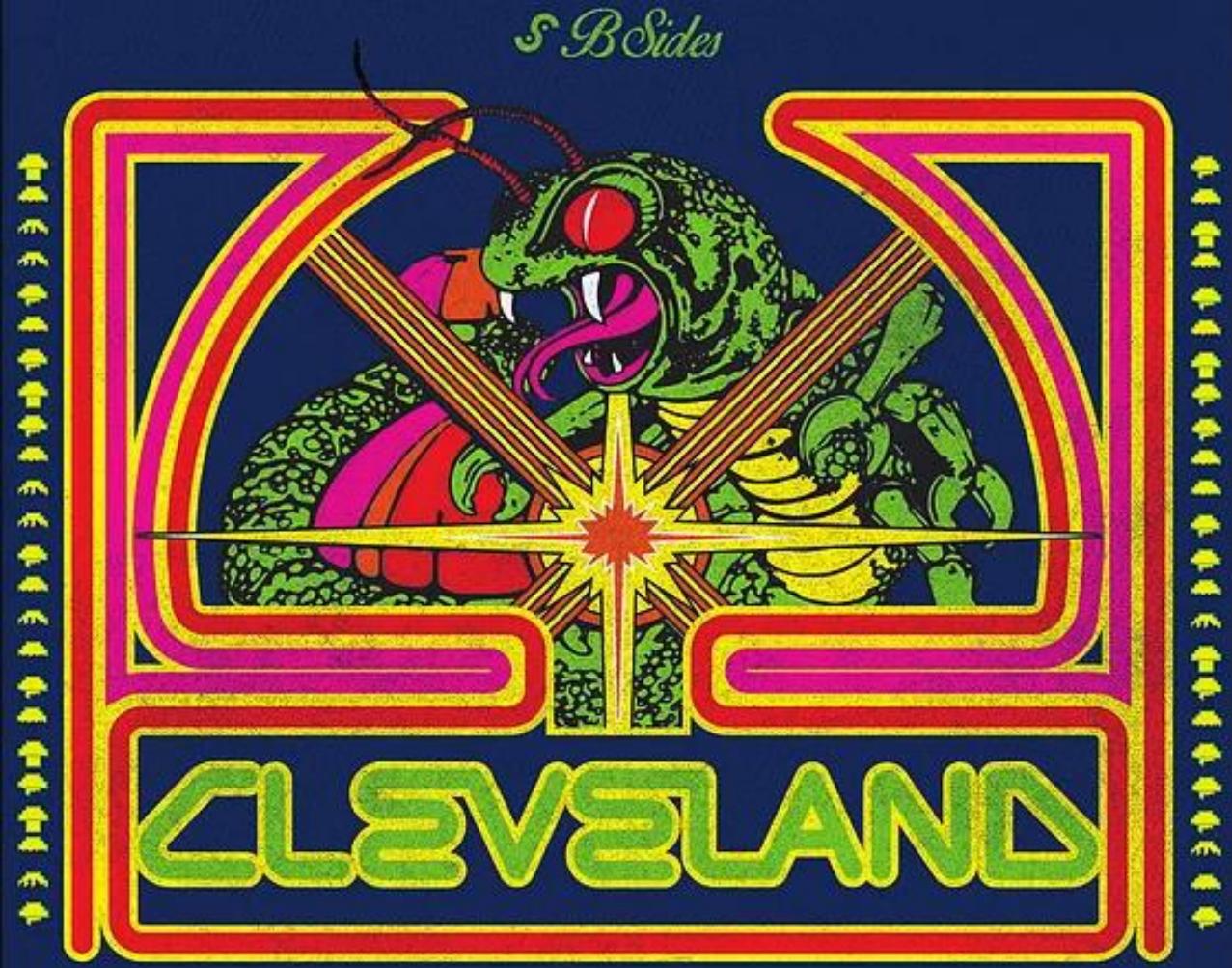




INFORMATION SECURITY +
ADVISORY SERVICES CONSULTING

Unix: The Other White Meat



#> whoami

Adam Compton

- Who Am I?
- Simple Answer:
 - Father/Husband/Son/Brother
 - Programmer/Pentester/Researcher
 - Hillbilly



#> whoami

David Boyd

- Pen Tester/Red Teamer
- Christian
- Husband
- Father
- US Army
- Geek, Gamer



```
#> cat agenda.txt
```

- DO:
 - Talk about Unix
 - Show Common Exploit Vectors
- NOT:
 - Teach all of Unix
 - Get in to debate of VI vs Emacs vs nano vs

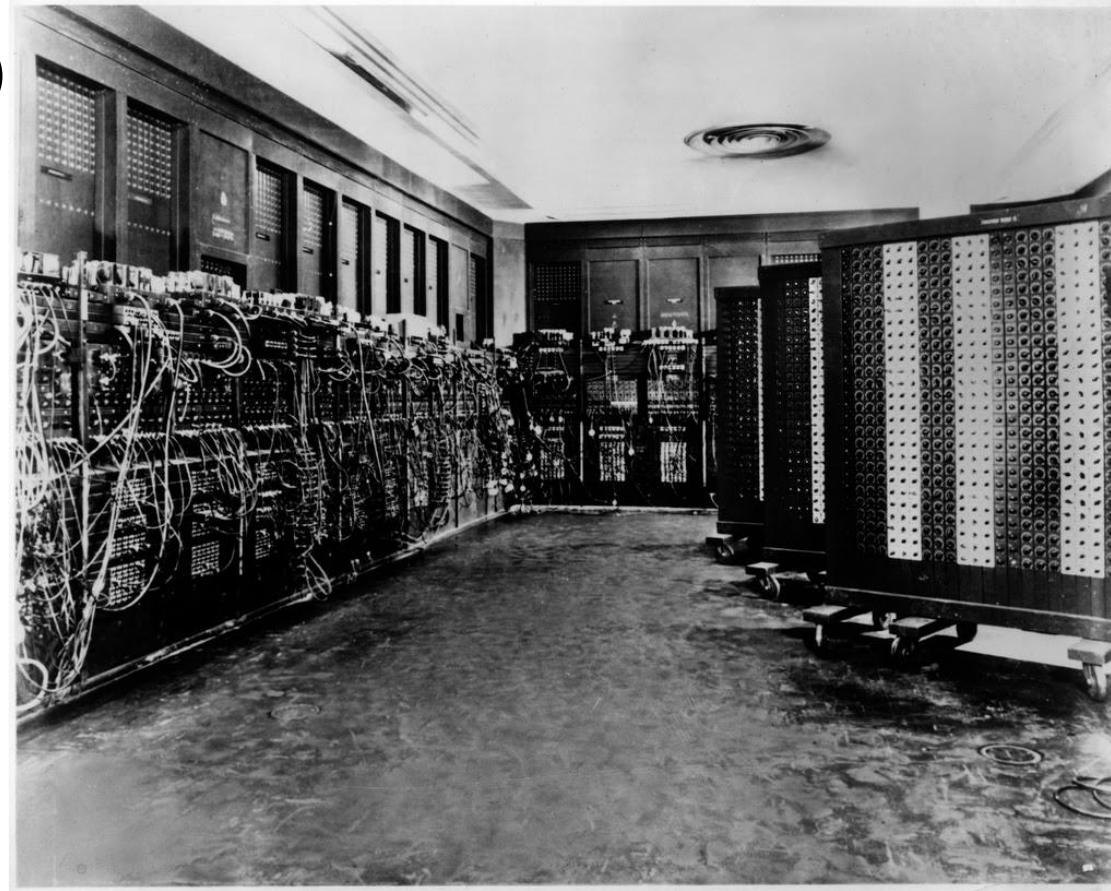


A Brief History of Unix



#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)



#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs

#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs
- 1969 - Ken Thompson & Dennis Ritchie and others start on what would become Unix



#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs
- 1969 - Ken Thompson & Dennis Ritchie and others start on what would become Unix
- 1971 - UNIX was developed for the PDP-11/20 written in Assembly



#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs
- 1969 - Ken Thompson & Dennis Ritchie and others start on what would become Unix
- 1971 - UNIX was developed for the PDP-11/20 written in Assembly
- 1973 - UNIX was completely rewritten in C



Brian W. Kernighan • Dennis M. Ritchie

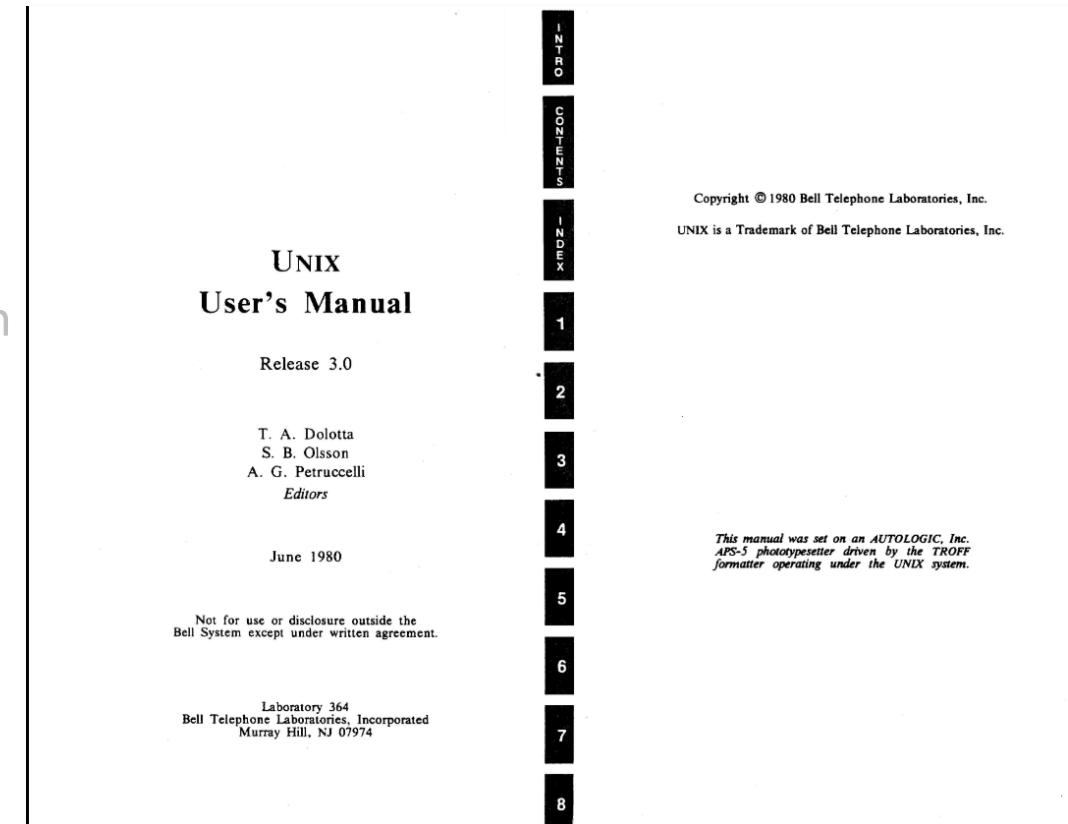
#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs
- 1969 - Ken Thompson & Dennis Ritchie and others start on what would become Unix
- 1971 - UNIX was developed for the PDP-11/20 written in Assembly
- 1973 - UNIX was completely rewritten in C
- 1978 - BSD is Released



#> history

- 1945 - ENIAC (1st* Large-Scale General Purpose Computer)
- 1964 - Multics (Multiplexed Information and Computing Service) developed by GE and Bell Labs
- 1969 - Ken Thompson & Dennis Ritchie and others start on what would become Unix
- 1971 - UNIX was developed for the PDP-11/20 written in Assembly
- 1973 - UNIX was completely rewritten in C
- 1978 - BSD is Released
- 1982 - AT&T's UNIS System Group releases System III, the first public release outside Bell Lab



#> history

- 1982 - SunOS 1.0, HP-UX, Ultrix-11



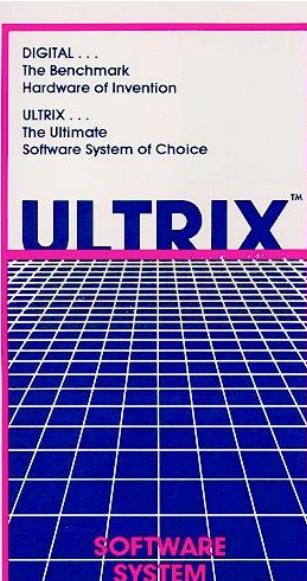
#> history

- 1982 - SunOS 1.0, HP-UX, Ultrix-11
- 1986 - AIX



#> history

- 1982 - SunOS 1.0, HP-UX, Ultix-11
- 1986 - AIX
- 1987 - IRIX

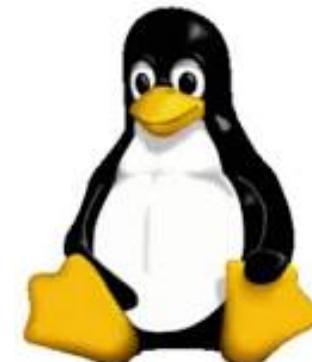
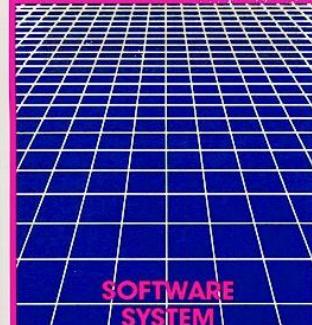


#> history

- 1982 - SunOS 1.0, HP-UX, Ultrix-11
- 1986 - AIX
- 1987 - IRIX
- 1991 - Solaris 1.0, Linus Torvalds starts on Linux 0.01



ULTRIX™



#> history

- 1982 - SunOS 1.0, HP-UX, Ultrix-11
- 1986 - AIX
- 1987 - IRIX
- 1991 - Solaris 1.0, Linus Torvalds starts on Linux 0.01
- 1995 - Digital Unix



DIGITAL . . .
The Benchmark
Hardware of Invention
ULTRIX . . .
The Ultimate
Software System of Choice

ULTRIX™

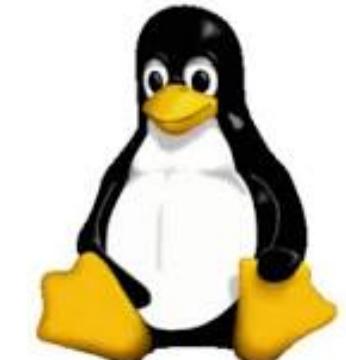
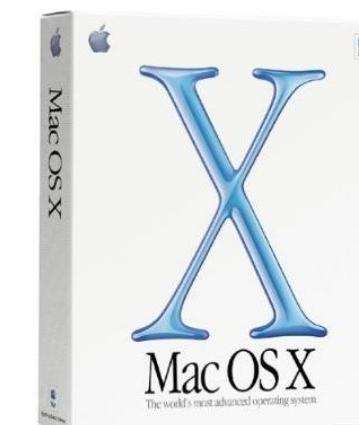
#> history

- 1982 - SunOS 1.0, HP-UX, Ultix-11
- 1986 - AIX
- 1987 - IRIX
- 1991 - Solaris 1.0, Linus Torvalds starts on Linux 0.01
- 1995 - Digital Unix
- 1999 - Tru64



#> history

- 1982 - SunOS 1.0, HP-UX, Ultix-11
- 1986 - AIX
- 1987 - IRIX
- 1991 - Solaris 1.0, Linus Torvalds starts on Linux 0.01
- 1995 - Digital Unix
- 1999 - Tru64
- 2001 - OSX 10 released



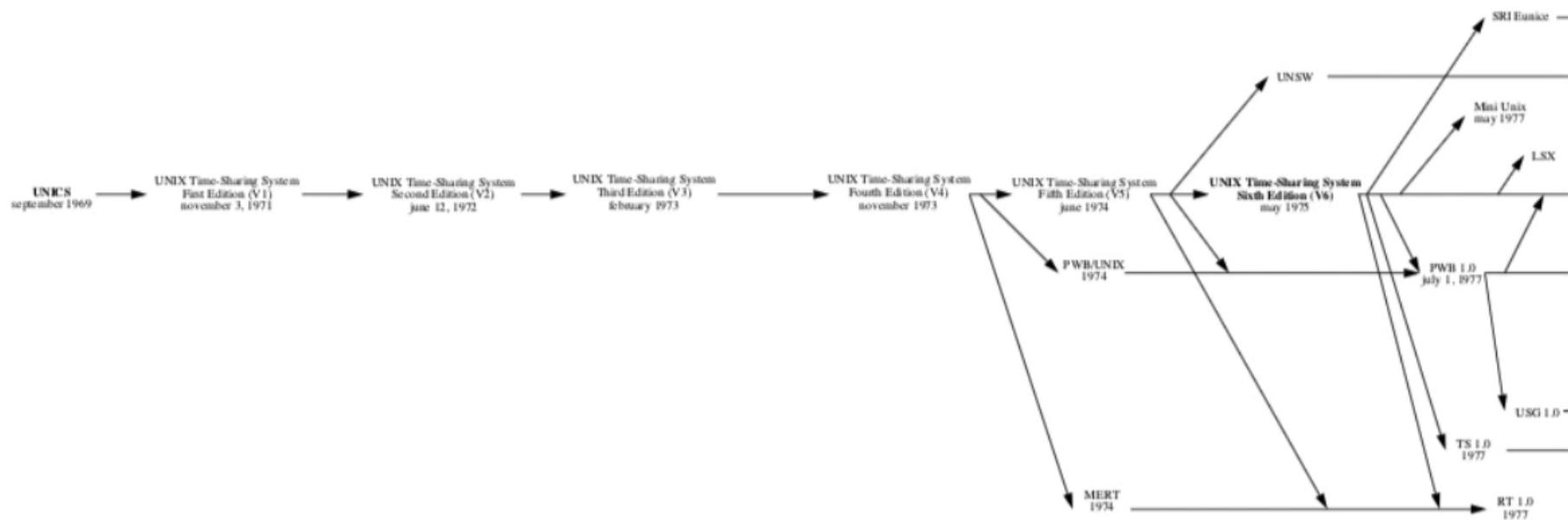
DIGITAL . . .
The Benchmark
Hardware of Invention

ULTRIX . . .
The Ultimate
Software System of Choice

ULTRIX™

1969 1970 1971 1972 1973 1974 1975 1976 1977

1969 1970 1971 1972 1973 1974 1975 1976 1977



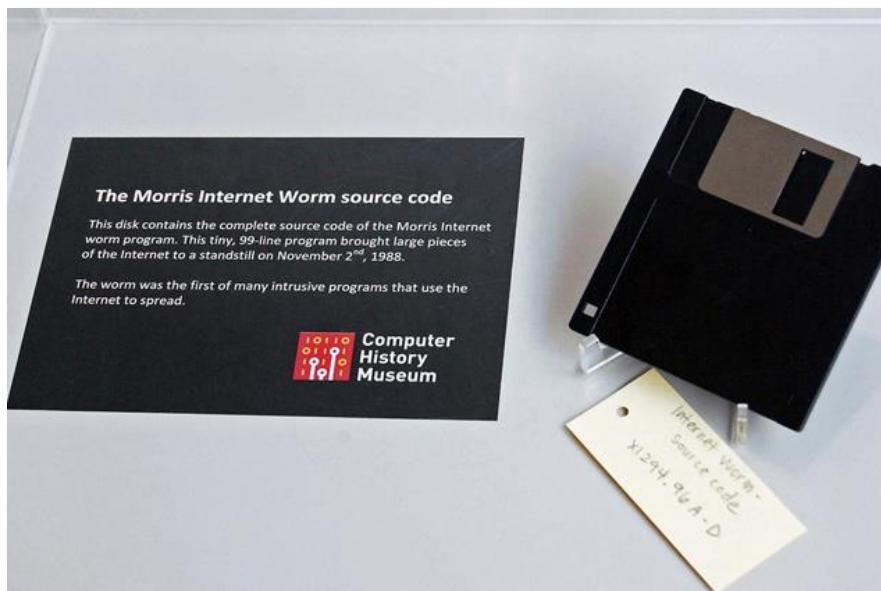
```
#> grep issues changelog.txt
```

- Passwords
- File Permissions
- Trust Relationships
- Application Flaws (BOF, HOF, etc...)



#> apropos "Morris Worm"

- Morris Worm (1988)
- targeted sendmail, finger, rsh/rexec, weak passwords
- Written by: Robert Tappan Morris @ Cornell University



```
#> cat /etc/passwd
```

/etc/passwd

<username>:<encrypted password>:<UID>:<GID>:<full name>:<home dir>:<shell>

<encrypted password> = crypt(plaintext password)

crypt() used to use DES (with a 12 bit number salt 0-4095)

salt selected based on time of day

converted to 2 character string and prepended to encrypted password

Improvements:

- /etc/shadow
- crypt() -> md5, sha1, sha256

#> ls -la

type	-	r w x	r w x	r w x
	-			
		owner	group	other
r	= normal file	= read access	= write access	= execute access
w	= directory	= 4	= 2	= 1
x	= symlink			
SETUID	= 4	-> "s"	in place of owner "x"; "S"	is SETUID and "x"
SETGID	= 2	-> "s"	in place of group "x"; "S"	if SETGID and "x"
STICKY	= 1	-> "t"	in place of other "x"; "T"	is STICKY and "x"



```
#> ls -la
```

Interesting files:

- /etc/password
- /var/log
- /var/mail
- .history
- .rhosts
- .forward
- "any ~/.xxxx File"

SETUID/SETGID Files

```
find / -perm /4000 -print  
find / -perm /2000 -print
```

```
#> echo "+ +" > ~/.rhosts
```

R-Commands

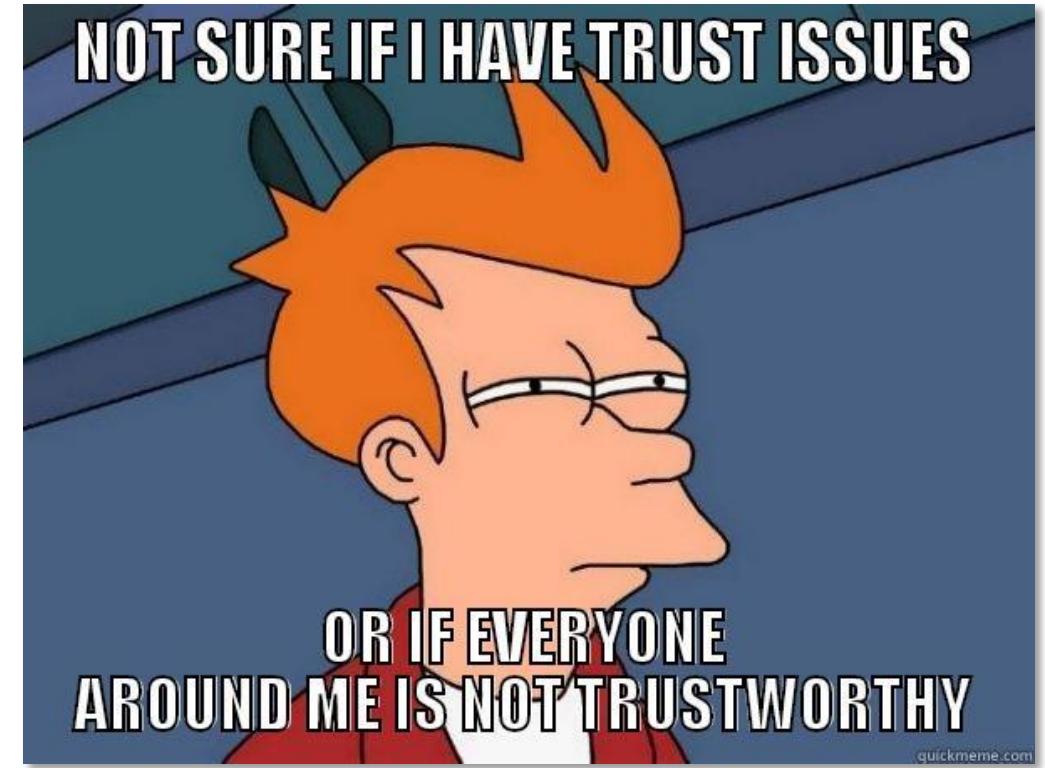
- rsh/rlogin/exec
- Important files:
 - ~/.rhosts
 - /etc/hosts.equiv

NFS

- showmount -e
- /etc/export

SUDO/su

- sudo -l



DEMO: R - Commands

The screenshot shows a terminal window with three tabs:

- Hack the Planet!
- root@metasploitable: ~
- firefart@Valentine: /tmp

The 'root@metasploitable' tab is active, displaying the command prompt [attacker@ubuntu] #.

DEMO: NFS Commands

The screenshot shows a terminal window with three tabs:

- Hack the Planet!
- root@metasploitable: ~
- firefart@Valentine: /tmp

The 'root@metasploitable' tab is active, displaying the prompt [attacker@ubuntu] #.

```
#> nmap -A -p- 10.0.0.0/24
```

Services:

- telnetd
- sadmind
- sendmail

Tools:

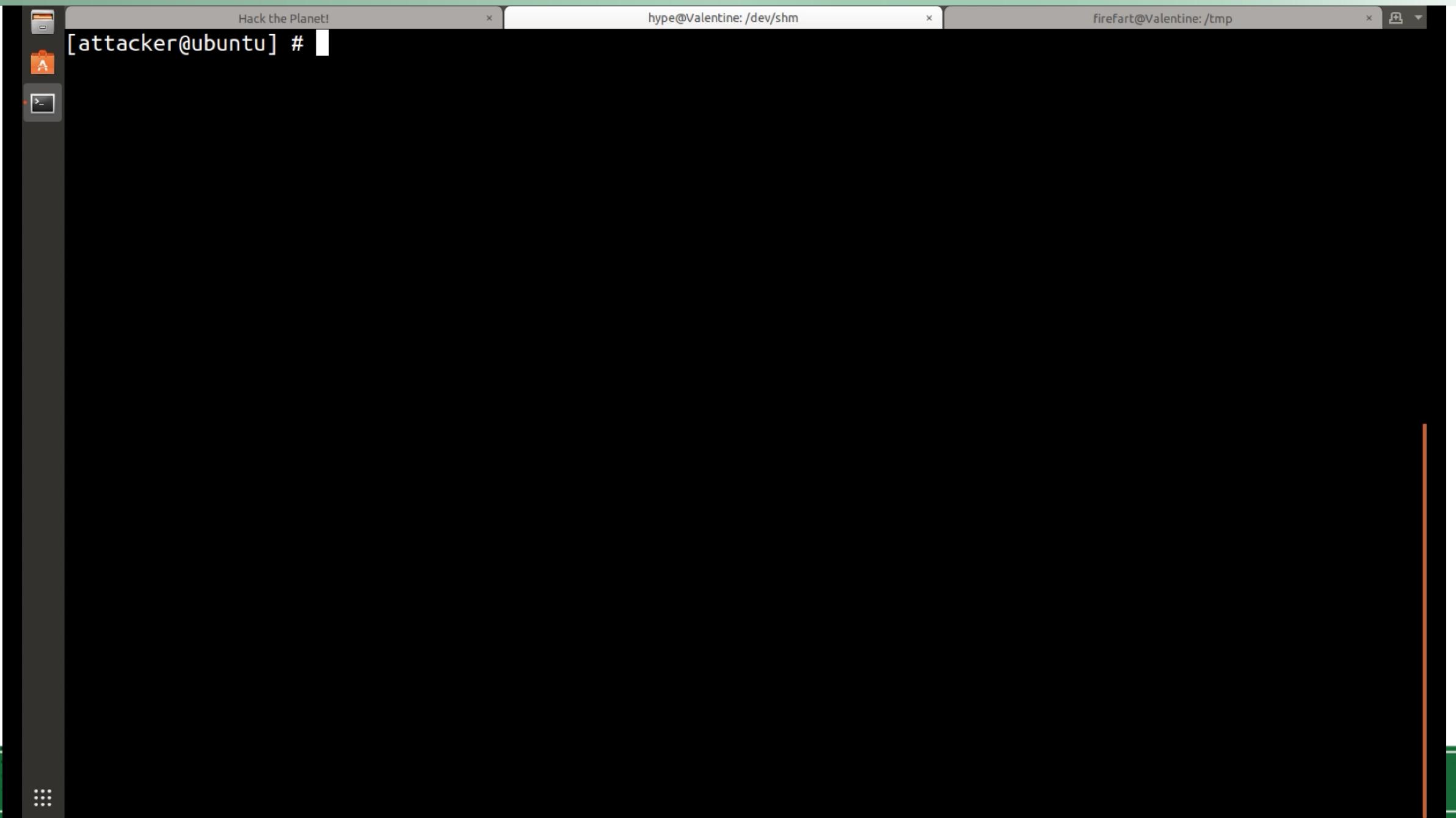
- Nmap

Web apps:

- phpmyadmin
- tomcat/jboss
- jenkins



DEMO: telnet -l "-fbin" x.x.x.x



Real World Privilege Escalation



```
#> echo "A Quick Note"
```



```
#> cat enum.txt
```

- Who am I? Who else is logged in? Who are superusers?
- What info do I have access to? What can I do?
- Where can I go?



```
#> cat example.txt
```

uname -a – Current kernel version

env – Current environment variable

pwd – Current directory

whoami – Current user

history – Command history for current user

cat ~/.bash_history – Bash history

sudo -l Commands you can run as sudo

cat /etc/sudoers – Who is in sudoers file

cat /etc/passwd – Additional users



```
#> cat EnumerationScripts.txt
```

LinEnum

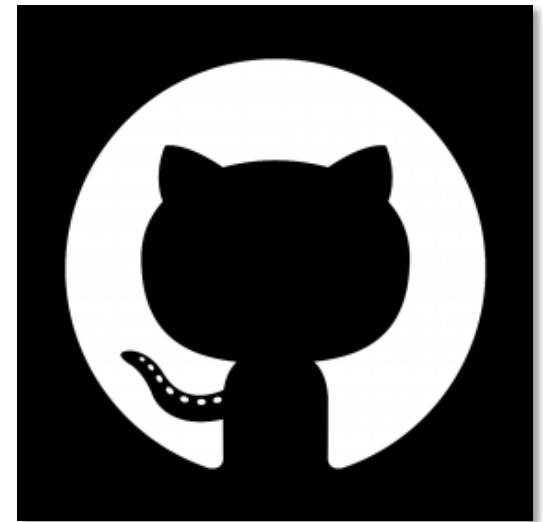
- <https://github.com/rebootuser/LinEnum>

LinuxPrivChecker

- <https://github.com/sleventyeleven/linuxprivchecker>

UnixPrivescCheck

- <https://github.com/pentestmonkey/unix-privesc-check>



DEMO: Enumeration Scripts

The image shows a Kali Linux desktop environment. In the top-left corner, there's a terminal window titled "File System" with the command "sec@shrek ~\$". To the right of the terminal is a large, semi-transparent watermark of the Kali Linux logo, which features a stylized green and red dragon head with the word "KALI" and "BY OFFENSIVE SECURITY" below it. The desktop background is dark. On the left side, there's a vertical dock with four icons: "Home" (house icon), "Trach" (camera icon), and "Kali Live" (target icon). At the bottom of the screen, there's a horizontal taskbar with four application icons: "File System", "Home", "Trach", and "Kali Live". The bottom status bar displays the user information "sec@shrek ~\$" and the date and time "22:40 27-Feb-19".



```
#> searchsploit -h
```



Searchsploit is a command line search tool for Exploit-DB that also allows you to take a copy of Exploit Database with you, everywhere you go.

SearchSploit gives you the power to perform detailed off-line searches through your locally checked-out copy of the repository.

Exploit Title	Path (/usr/share/exploitdb/)
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Huge Zero Page (1)	exploits/linux/dos/43199.c
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Huge Zero Page (2)	exploits/linux/dos/44305.c
Linux Kernel 2.6.22 < 3.9 (x86/x64) - ' Dirty COW /proc/self/mem' Race Condition Privilege Escalation (SUID Method)	exploits/linux/local/40616.c
Linux Kernel 2.6.22 < 3.9 - ' Dirty COW /proc/self/mem' Race Condition Privilege Escalation (/etc/passwd Method)	exploits/linux/local/40847.cpp
Linux Kernel 2.6.22 < 3.9 - ' Dirty COW PTRACE_POKEDATA' Race Condition (Write Access Method)	exploits/linux/local/40838.c
Linux Kernel 2.6.22 < 3.9 - ' Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege Escalation (/etc/passwd Method)	exploits/linux/local/40839.c
Linux Kernel 2.6.22 < 3.9 - ' Dirty COW' /proc/self/mem Race Condition (Write Access Method)	exploits/linux/local/40611.c
Quick and Dirty Blog (qdblog) 0.4 - 'categories.php' Local File Inclusion	exploits/php/webapps/4603.txt
Quick and Dirty Blog (qdblog) 0.4 - SQL Injection / Local File Inclusion	exploits/php/webapps/3729.txt
snapd < 2.37 (Ubuntu) - ' dirty_sock ' Local Privilege Escalation (1)	exploits/linux/local/46361.py
snapd < 2.37 (Ubuntu) - ' dirty_sock ' Local Privilege Escalation (2)	exploits/linux/local/46362.py
Shellcodes: No Result	

```
#> find / -perm /4000 -ls
```

SUID (Set owner User ID up on execution) is a special type of file permissions given to a file. Normally in Linux/Unix when a program runs, it inherits access permissions from the logged in user. SUID is defined as giving temporary permissions to a user to run a program/file with the permissions of the file owner rather than the user who runs it.

In simple words, users will get file owners permissions as well as owner UID and GID when executing a file/program/command.

DEMO: SETUID

The screenshot shows a terminal window with three tabs open:

- msfadmin@metasploitable:~\$**: This tab is active and shows the user msfadmin on a Metasploitable host.
- root@metasploitable:~**: This tab shows the root user on the same host.
- Firefart@Valentine: /tmp**: This tab shows a user named Firefart on a host named Valentine, located in the /tmp directory.

The terminal interface includes a vertical toolbar on the left with icons for file operations (File, New, Open, Save, Copy, Paste) and a bottom navigation bar with a grid icon and a small logo in the bottom right corner.

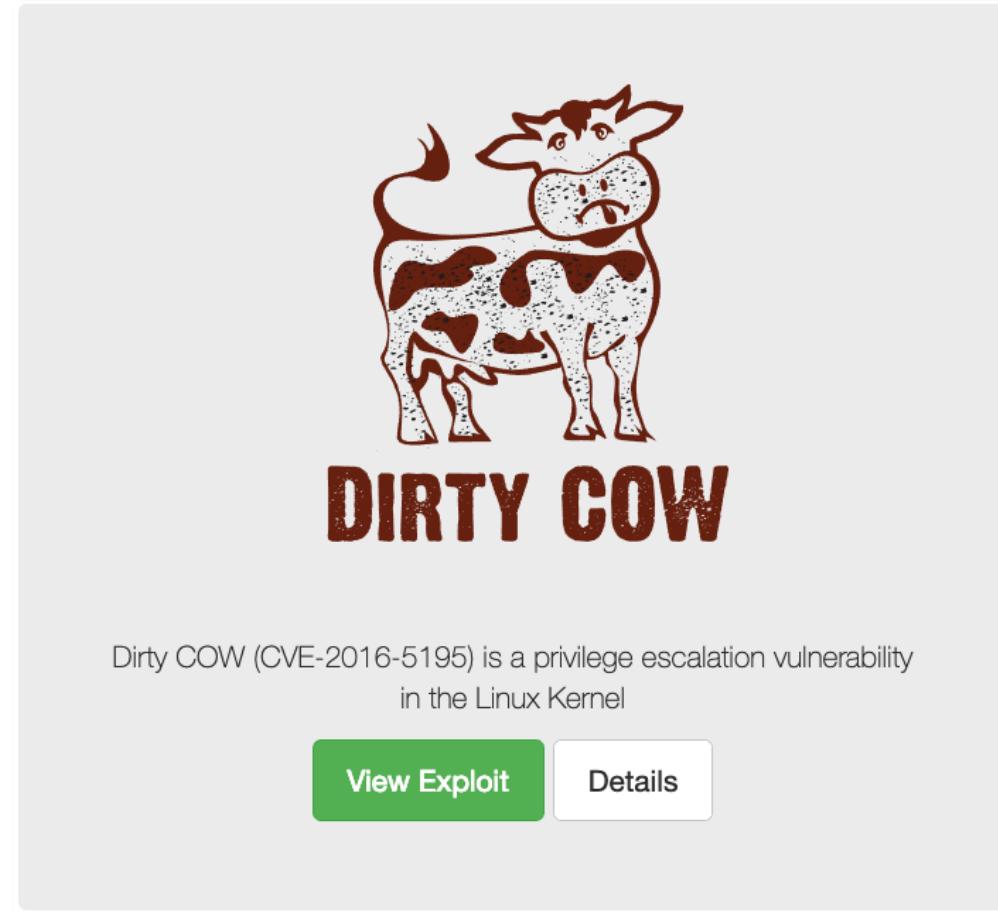
#> ls /boot/vmlinuz*

- Mempodipper
- DirtyCow
- sendpage
- Dirty_Sock

```
Linux Kernel 2.4/2.6 - 'sock_sendpage()' Local Privilege Escalation (3) | exploits/linux/local/9641.txt
Linux Kernel 2.6 (Debian 4.0 / Ubuntu / Gentoo) UDEV < 1.4.1 - Local Privilege Escalation (1) | exploits/linux/local/8478.sh
Linux Kernel 2.6 (Gentoo / Ubuntu 8.10/9.04) UDEV < 1.4.1 - Local Privilege Escalation (2) | exploits/linux/local/8572.c
Linux Kernel 2.6 < 2.6.19 (White Box 4 / CentOS 4.4/4.5 / Fedora Core 4/5/6 x86) - 'ip_append_data()' Ring0 Privilege Escala| exploits/linux/x86/local/9542.c
Linux Kernel 2.6.0 < 2.6.31 - 'pipe.c' Local Privilege Escalation (1) | exploits/linux/local/33321.c
Linux Kernel 2.6.10 < 2.6.31.5 - 'pipe.c' Local Privilege Escalation | exploits/linux/local/40812.c
Linux Kernel 2.6.13 < 2.6.17.4 - 'logrotate prctl()' Local Privilege Escalation | exploits/linux/local/2031.c
Linux Kernel 2.6.13 < 2.6.17.4 - 'sys_prctl()' Local Privilege Escalation (1) | exploits/linux/local/2004.c
Linux Kernel 2.6.13 < 2.6.17.4 - 'sys_prctl()' Local Privilege Escalation (2) | exploits/linux/local/2005.c
Linux Kernel 2.6.13 < 2.6.17.4 - 'sys_prctl()' Local Privilege Escalation (3) | exploits/linux/local/2006.c
Linux Kernel 2.6.13 < 2.6.17.4 - 'sys_prctl()' Local Privilege Escalation (4) | exploits/linux/local/29714.txt
Linux Kernel 2.6.17 < 2.6.24.1 - 'Sys Tee' Local Privilege Escalation | exploits/linux/local/29714.txt
Linux Kernel 2.6.17 < 2.6.24.1 - 'vmslice' Local Privilege Escalation (2) | exploits/linux/local/5092.c
Linux Kernel 2.6.17 < 2.6.24.1 - 'proc' Local Privilege Escalation | exploits/linux/local/2013.c
Linux Kernel 2.6.18 < 2.6.18-20 - Local Privilege Escalation | exploits/linux/local/10613.c
Linux Kernel 2.6.22 < 3.9 (x86/x64) - 'Dirty COW /proc/self/mem' Race Condition Privilege Escalation (SUID Method) | exploits/linux/local/40616.c
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW /proc/self/mem' Race Condition Privilege Escalation (/etc/passwd Method) | exploits/linux/local/40847.cpp
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege Escalation (/etc/passwd Method) | exploits/linux/local/40839.c
Linux Kernel 2.6.23 < 2.6.24 - 'vmslice' Local Privilege Escalation (1) | exploits/linux/local/5093.c
Linux Kernel 2.6.24_16-23/2.6.27_7-10/2.6.28.3 (Ubuntu 8.04/8.10 / Fedora Core 10 x86-64) - 'set_selection()' UTF-8 Off-by-0 | exploits/linux/x86-64/local/9083.c
Linux Kernel 2.6.27 < 2.6.36 (RedHat x86-64) - 'compat' Local Privilege Escalation | exploits/linux/x86-64/local/15024.c
Linux Kernel 2.6.28/3.0 (DEC Alpha Linux) - Local Privilege Escalation | exploits/linux/local/17391.c
Linux Kernel 2.6.29 - 'ptrace_attach()' Race Condition Privilege Escalation | exploits/linux/local/8678.c
Linux Kernel 2.6.30 < 2.6.30.1 / SELinux (RHEL 5) - Local Privilege Escalation | exploits/linux/local/9191.txt
Linux Kernel 2.6.32 (Ubuntu 10.04) - '/proc' Handling SUID Privilege Escalation | exploits/linux/local/41770.txt
Linux Kernel 2.6.32 - 'pipe.c' Local Privilege Escalation (4) | exploits/linux/local/10018.sh
Linux Kernel 2.6.32 < 3.x (CentOS 5/6) - 'PERF EVENTS' Local Privilege Escalation (1) | exploits/linux/local/25444.c
Linux Kernel 2.6.36-rc8 - 'RDS Protocol' Local Privilege Escalation | exploits/linux/local/15285.c
Linux Kernel 2.6.37 (RedHat / Ubuntu 10.04) - 'Full-Nelson.c' Local Privilege Escalation | exploits/linux/local/15704.c
Linux Kernel 2.6.39 < 3.2.2 (Gentoo / Ubuntu x86/x64) - 'Mempodipper' Local Privilege Escalation (1) | exploits/linux/local/18411.c
Linux Kernel 2.6.39 < 3.2.2 (x86/x64) - 'SYS_EPOLL_WAIT' Local Integer Overflow / Local Privilege Escalation | exploits/linux/local/35161.c
Linux Kernel 2.6.9 < 2.6.11 (RHEL 4) - 'SYS_EPOLL_WAIT' Local Integer Overflow / Local Privilege Escalation | exploits/linux/local/1397.c
Linux Kernel 2.6.x (Gentoo 2.6.29rc1) - 'ptrace_attach' Local Privilege Escalation (1) | exploits/linux/local/8673.c
Linux Kernel 2.6.x - 'pipe.c' Local Privilege Escalation (2) | exploits/linux/local/25202.c
Linux Kernel 2.6.x - Ext4 'move extents' ioctl Privilege Escalation | exploits/linux/local/33395.txt
Linux Kernel 2.6.x - Ptrace Privilege Escalation | exploits/linux/local/30604.c
Linux Kernel 2.6.x / 3.10.x / 4.14.x (RedHat / Debian / CentOS) (x64) - 'Mutagen Astronomy' Local Privilege Escalation | exploits/linux/local/45516.c
Linux Kernel 2.x (Android) - 'sock_sendpage()' Local Privilege Escalation | exploits/android/local/9477.txt
Linux Kernel 2.x (RedHat) - 'sock_sendpage()' Ring0 Privilege Escalation (1) | exploits/linux/local/9435.txt
Linux Kernel 2.x - 'sock_sendpage()' Local Privilege Escalation (4) | exploits/linux/local/9436.txt
Linux Kernel 3.0 < 3.3.5 - 'CLONE_NEWUSER|CLONE_FS' Local Privilege Escalation | exploits/linux/local/38390.c
```



#> apropos DirtyCow



```
#> apt-get update
```

Ubuntu

- 4.8.0-26.28 for Ubuntu 16.10
- 4.4.0-45.66 for Ubuntu 16.04 LTS
- 3.13.0-100.147 for Ubuntu 14.04 LTS
- 3.2.0-113.155 for Ubuntu 12.04 LTS

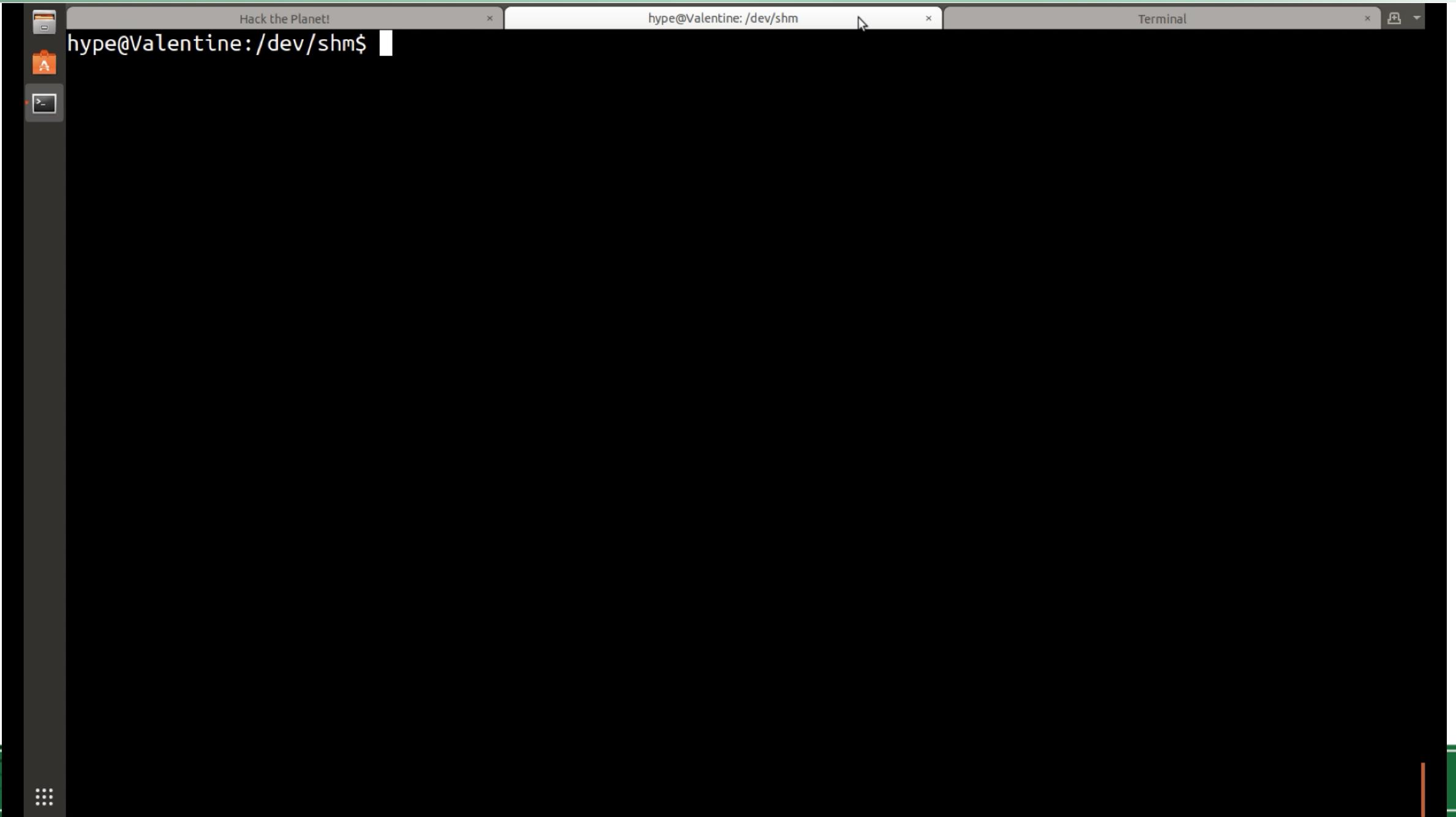
Debian

- 3.16.36-1+deb8u2 for Debian 8
- 3.2.82-1 for Debian 7
- 4.7.8-1 for Debian unstable

Arch

- 4.4.26-1 for ArchLinux (linux-lts package)
- 4.8.3 for ArchLinux (linux package)

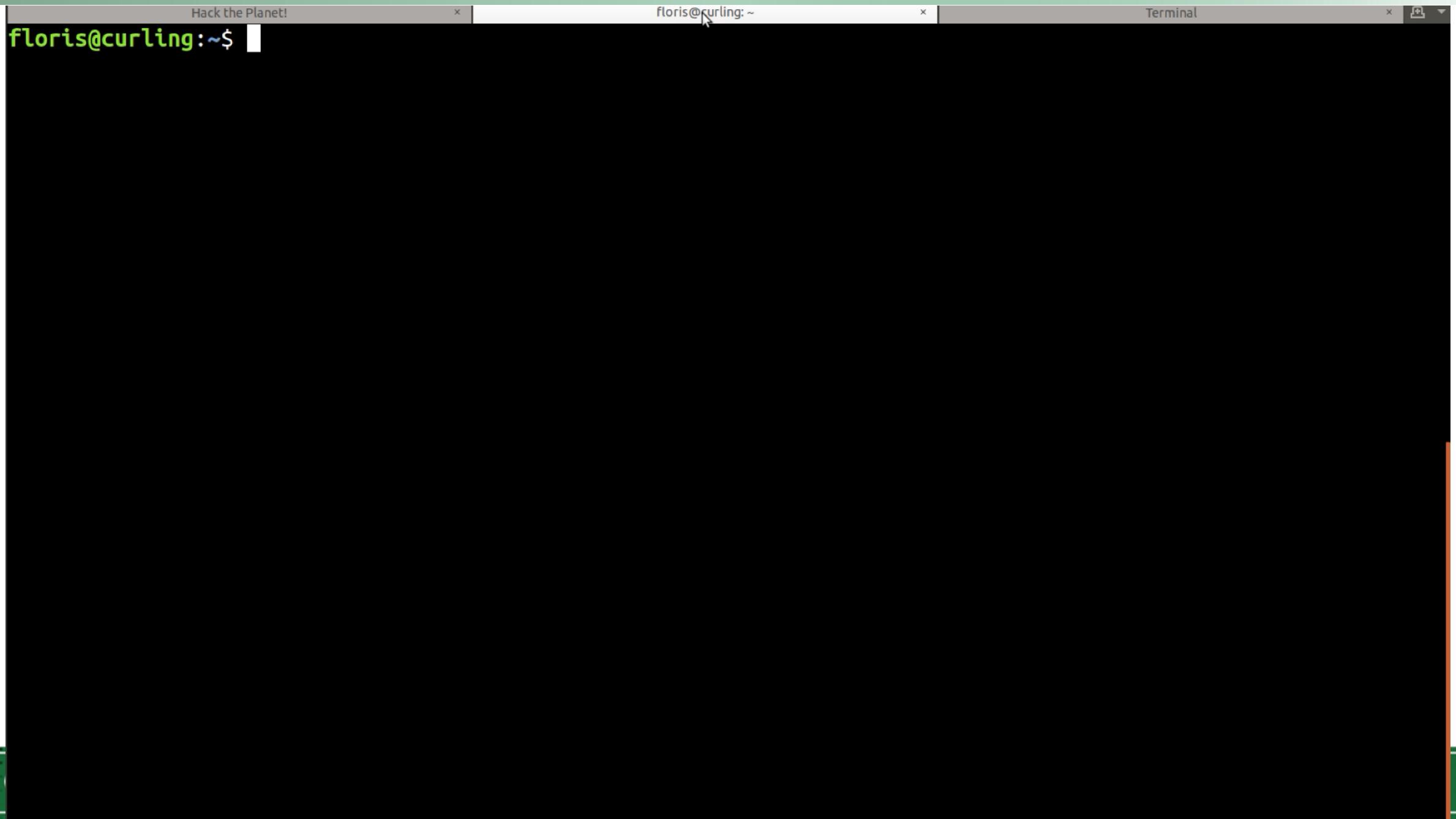
DEMO: DirtyCow



#> apropos DirtySock

- <https://initblog.com/2019/dirty-sock/>
- A privilege escalation vulnerability in default installations of Ubuntu Linux. This was due to a bug in the ‘snapd’ API, a default service. Any local user could exploit this vulnerability to obtain immediate root access to the system.
- Affects snapd versions < 2.37.1

DEMO: DirtySock



Final Thoughts



#> cat Additional_Resources.txt

Basic Linux Privilege Escalation – G0tmi1k

- <https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/>

BSides Augusta – Jake Williams

- <https://www.youtube.com/watch?v=dk2wsyFiosg>

Tradecraft Security Weekly

- <https://www.youtube.com/watch?v=oYHAI0cgur4>

```
#> echo "Try Harder"
```

HackTheBox

- <https://www.hackthebox.eu/>



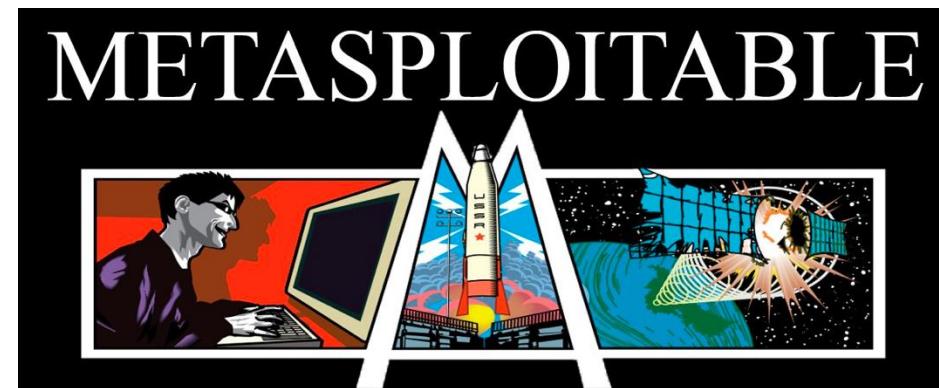
VulnHub

- <https://www.vulnhub.com/>



Metasploitable

- <https://information.rapid7.com/download-metasploitable-2017.html>



Contact Us!

Adam L. Compton

 @tatanus

 www.hillbillystorytime.com

 www.youtube.com/hillbillstorytime

 adam.comptom@gmail.com

 adam.compton@trustedsec.com

David R. Boyd

 @fir3d0g

 www.twitch.tv/fir3d0g

 fir3d0g

 techboyd@gmail.com

 david.boyd@trustedsec.com