Information Security Strategy & Tactics in the Win32 Space

Information Security Strategy & Tactics in the Win32 Space

Craig Chamberlain
Principal Technical Consultant
Verdasys, Inc.

The Problems

- Software is a complex system that cannot be perfected by humans who do not produce error-free code.
- Computers are very sophisticated idiots; software based security technologies are no match for a human mind.

The Players

IT: “Management has neither the ability or the inclination to fully grasp the technical complexity of the challenges we face. They continue to increase expectations while reducing resources and and creating unfunded mandates.”

The Players

- Management: “IT doesn’t seem to grasp the low value proposition they offer in the area of security; they can offer no assurance that a targeted attack can be defended against while they clamor for the allocation of more resources, excitedly pointing out their ability to deal with low or medium-intensity threats, but possibly at a cost that exceeds a single loss expectancy..”

The Players

- Your adversaries are more numerous. Some are more experienced and some are more intelligent.
- The advantage is theirs; you must defend everywhere; they need only find a single route of penetration you have overlooked.
- They will tend to be highly proficient in one or two technical environments but may display an impressive ability to learn as they go.
- They may tend to repeat their methods with which they have had past success.

Strategy

- Signature matching technologies are effective against threats that have a known signature.
- Access controls are effective so long as they are not compromised or abused.
- Patching protects hosts against known vulnerabilities.

Strategy

Choose your battles and position your pieces thoughtfully.

Plan to fail (but congratulations, failure is one of the basic freedoms and provides a rich learning experience).

Learn to identify the routes of penetration that your adversaries may use. “If you know the enemy and know yourself, You need not fear the result of a hundred battles.”

-Sun Tzu , *Art of War*

Strategy

Determine your security policy:
what are you trying to achieve?

Business continuity

Intellectual property

Regulatory compliance

Strategy

- Design defenses to minimize attack surfaces and potential routes of penetration.
- Design perimeter defenses with the assumption Internet-facing systems will be compromised.
- Design interior access controls with the assumption they will be abused.

Attack Profiles

41%	Known OS vulnerability
26%	Known application vulnerability
20%	Insider access
17%	Poor access control
15%	Unknown OS vulnerability
10%	Guessed Passwords
6%	Unknown application

Source: GISS, 2001, sample size: 8100 (multiple responses)

41% Known OS vulnerability
26% Known application vulnerability

- Frequently exploited by invasive self-replicating mobile code
- Viruses, worms
- Trojan Horses
- Backdoors
- Rootkits

“If J. Random Websurfer clicks on a button that promises dancing pigs on his computer monitor, and instead gets a hortatory message describing the potential dangers of the applet - he’s going to choose dancing pigs over computer security any day. If the computer prompts him with a warning screen like: “The applet DANCING PIGS could contain malicious code that might do permanent damage to your computer, steal your life’s savings, and impair your ability to have children,” he’ll click “OK” without even reading it. Thirty seconds later he won’t even remember that the warning screen even existed.”

- Bruce Schneier, *Secrets & Lies. Digital Security in a Networked World*

Invasive Code

Outlook SR-1 Update

E-mail attachment security prevents users from accessing several file types when sent as e-mail attachments. Affected file types include executables, batch files, and other file types that contain executable code often used by malicious hackers to spread viruses.

Object Model Guard prompts users with a dialog box when an external program attempts to access their Outlook Address Book or send e-mail on their behalf.

-from Microsoft Office Online

Invasive Code

- Are you interested in policing every desktop? Do you have control of every desktop?
- Consider more holistic approaches; attachments can be stripped before reaching the user
- Question assumptions; what is the business case for receiving executable attachments?
- Security policy can be enforced on remote access clients w/ Server 2003 “quarantine”
- Clients that roam to foreign networks need host-level defenses. Windows XP has a host level firewall.

Invasive Code

- When dealing with self-replicating mobile code that moves at wire speed, 99% effectiveness may return the same end result as 1% effectiveness.
- Antivirus software needs to be universally deployed and highly efficient; organizations that are still walking around managing antivirus configurations are going to spend their security resources on virus incidents.
- Potential routes of penetration should be identified so they can be closed during a major outbreak. First responders need to be empowered to take this action.
- Dust off those business continuity plans…

Browser Based Invasive Code

- Internet Explorer configuration can be managed w/ group policy or the Internet Explorer Administration Kit
- Can you reach 100% policy compliance at the desktop?
- ActiveX, Java can also be filtered at network perimeter

Patch Management

Develop a patch management strategy. Achieving patch compliance can be resource intensive. Prioritize exposed services for patching.

Patch management can and should be semi-automated. Tools include:

Microsoft Baseline Security Analyzer

Microsoft Software Update Services

Network World review of patch management tools

Known OS / Application vulnerability

- Internet-facing servers need least-privilege network level inbound and outbound access control.
- Intrusion detection can spot common exploits in action and possibly stop them. IDS systems needs attention and are not useful if nobody is paying attention.
- Intrusion prevention technologies (e.g. Cisco Security Agent) can pick up where these leave off to an extent.

Known OS / Application vulnerability

- Vulnerability / penetration testing can be valuable if it is an ongoing, heuristic process. If testing is a one-time event the value here is questionable and resources may be better spent elsewhere.
- Vulnerability assessment is only accurate when performed *before* a compromise.
- Requires significant resource allocation as this is more effective when repeated indefinitely.
- Core Security in Boston has a nice pentest tool:
- http://www2.corest.com/products/coreimpact/index.php

Unauthorized Access

- Terminal Server has a very good security history.
- There is no patch for the brute-force attack. Use high-entropy passwords, rename administrator accounts, enable administrator account lockout with passprop (NT 4) and admnlock (Win2KK), enable auditing of logon and logoff events. These two tools are found in the resource kits.
- Restrict remote management connections with access lists: permit ip 209.58.x.x /24 host tserver
- Perform remote management over VPNs.
- Tools from http://www.foundstone.com
- Ntlast.exe – command line tool that returns failed logins for an account
- VisualLast – centralized security log collection & review

Passwords

NTLM / NTLM v2 Authentication Info

http://support.microsoft.com/default.aspx?scid=kb;en-us;239869

http://support.microsoft.com/default.aspx?scid=kb;EN-US;147706

SAM (user account) encryption key can be exported:
http://support.microsoft.com/default.aspx?scid=kb;en-us;310105&Product=win2000

17%	Poor access Control
15%	Unknown OS Vulnerability

Internet Information Server

Tools for host hardening:

IIS Lockdown tool

URLscan

Account used for anonymous access does not need to the “login interactively” privilege

Disable unneeded script mappings

Delete binaries that can be used for remote compromise (cmd.exe, tftp, telnet) and filter outbound traffic

Place default directories on a non-system NTFS drive like D:\ and set file level ACLs to read-only

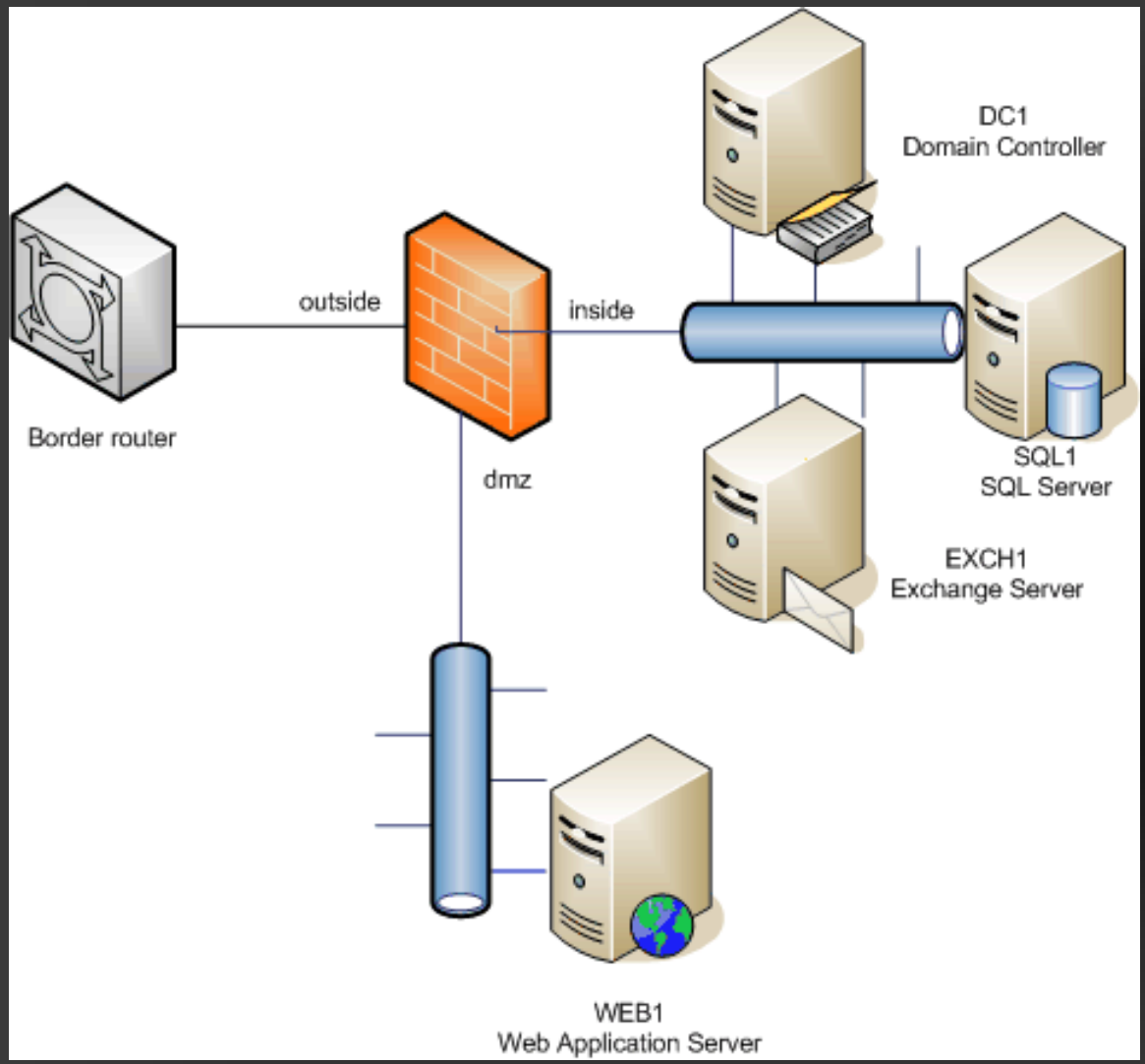
Host level intrusion detection (e.g. Tripwire)

Keep a system disk image warmed up (CD in the CD-ROM drive, boot floppy)

SQL Server

- Is there a business case for exposing the SQL port to public networks?
- Access-lists and VPNs are available (effort, yes, but probably easier than SQL injection testing)
- Remove xp_cmdshell (can be used to run shellcode)
- SQL does not require the context of the local system account
- Use non-privileged accounts for web application access
- Test for SQL injection in web applications
- Stored Procedures
- IPsec filtering
- Intrusion Detection
- Auditing & logging

17%	Poor access Control
15%	Unknown OS Vulnerability



Before

Access-list outside

Permit ip any host EXCH1 eq 25

Permit ip any host EXCH1 eq 80

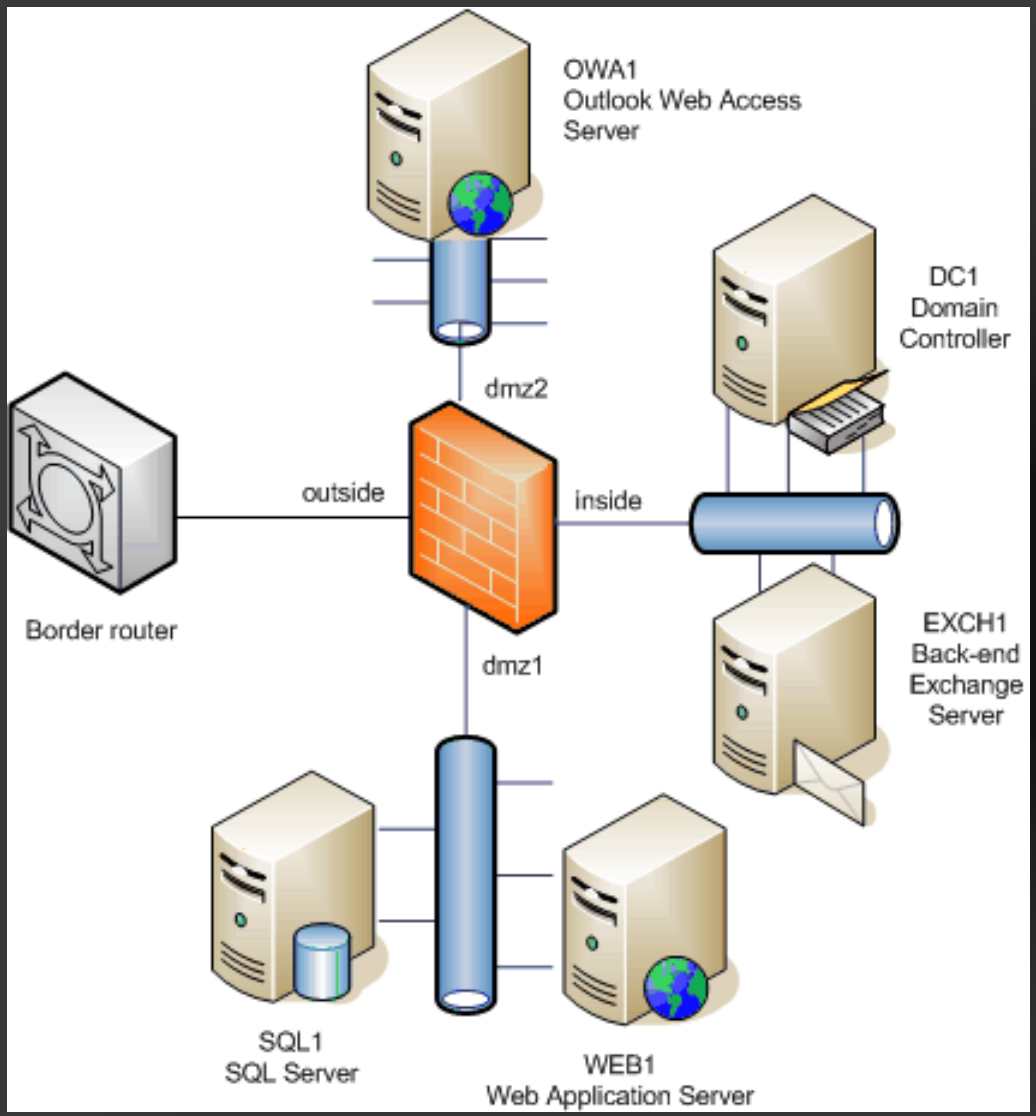
Permit ip any host WEB1 eq 80

Permit ip any host WEB1 eq 3389

Permit ip any host DC1 eq pptp

Access-list dmz

Permit ip any any



After

Access-list outside
Permit ip any host OWA1 eq 25
Permit ip any host OWA1 eq 443
Permit ip any host WEB1 eq 80
permit ip any host WEB1 eq 3389

Acces-list dmz2
Permit ip host OWA1 host EXCH1 eq ldap
Permit ip host OWA1 host EXCH1 Eq 25

Access-list inside

Permit ip inside dmz1 eq any

Permit ip inside dmz2 eq any

Permit ip host EXCH1 any eq SMTP

Permit ip inside any eq DNS

Permit ip inside any eq 80

Permit ip inside any eq 443

Network Access Control

- IPSEC filters between OWA1 and EXCH1
- User certificates or group passwords for VPN, wireless authentication
- Smart cards for privileged accounts

Event Correlation

According to Marcus Ranum, an independent computer and communications security consultant in Woodbine, Md., "Correlation is something everyone wants, but nobody even knows what it is. it's like liberty or free beer -- everyone thinks it's a great idea and we should all have it, but there's no road map for getting from here to there.“

July 28, 2003 (Computerworld)

Will be the future of information security as second and third generation tools appear.

Local company doing this: Q1 Labs, Inc.

http://www.q1labs.com/

6% Unknown application

"In the end, determining if an arbitrary piece of code will behave maliciously in advance of executing it is as difficult as the Halting problem." - Gary McGraw and Greg Morrisett, *Report to the Infosec Research Council on Malicious Code*

6% Unknown application: rootkits

- Rootkits are getting more sophisticated in the win32 space
- Process hiding, execution redirection, stack manipulation
- Some are detected by antivirus scanners
- Kernel file patches can be detected with host IDS (e.g. Tripwire)

Rootkits: How to Detect?

- Can’t hide their traffic at the network layer; many make or receive network connections
- Behavior-matching host-level intrusion prevention technology (e.g. Cisco Security Agent) can block system calls used by rootkits

6% Unknown application

"Well, you'll never get in through the front-line security but you might look for a backdoor".

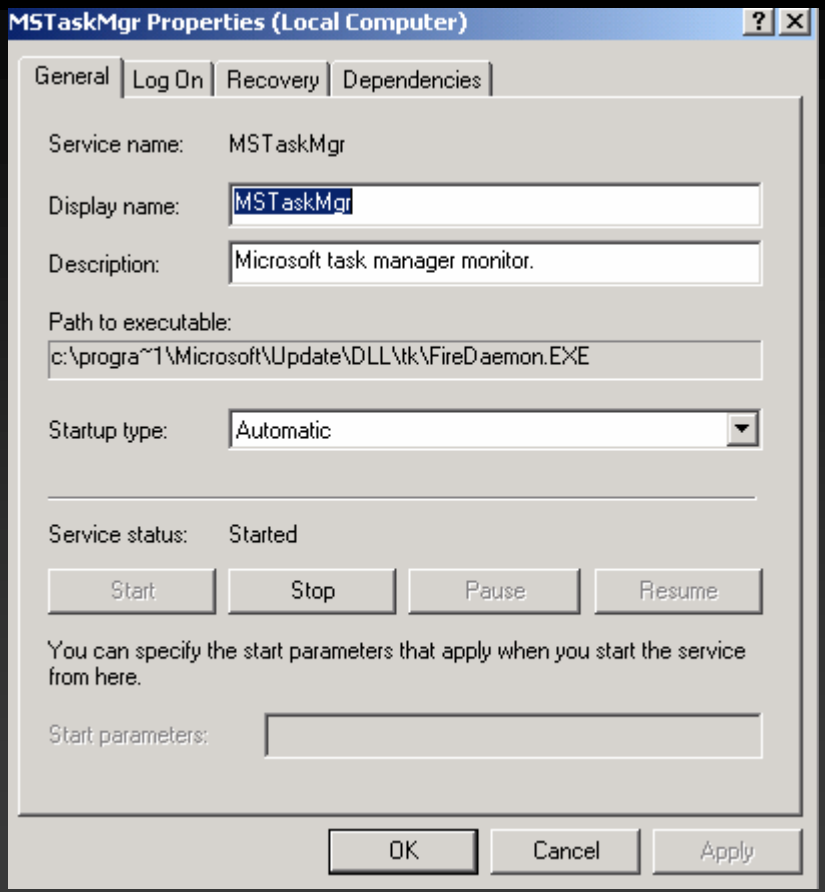
-Jim in *Wargames*

Forensic Tools

Active Ports from Smartline, Inc
Autorun from Sysinternals
FIRE bootable CD-ROM
Fport by Foundstone
Prcview from Igor Nys
Process Explorer by Sysinternals
Pulist from the Windows 2000 Resource Kit
TCPView by Sysinternals
Windump

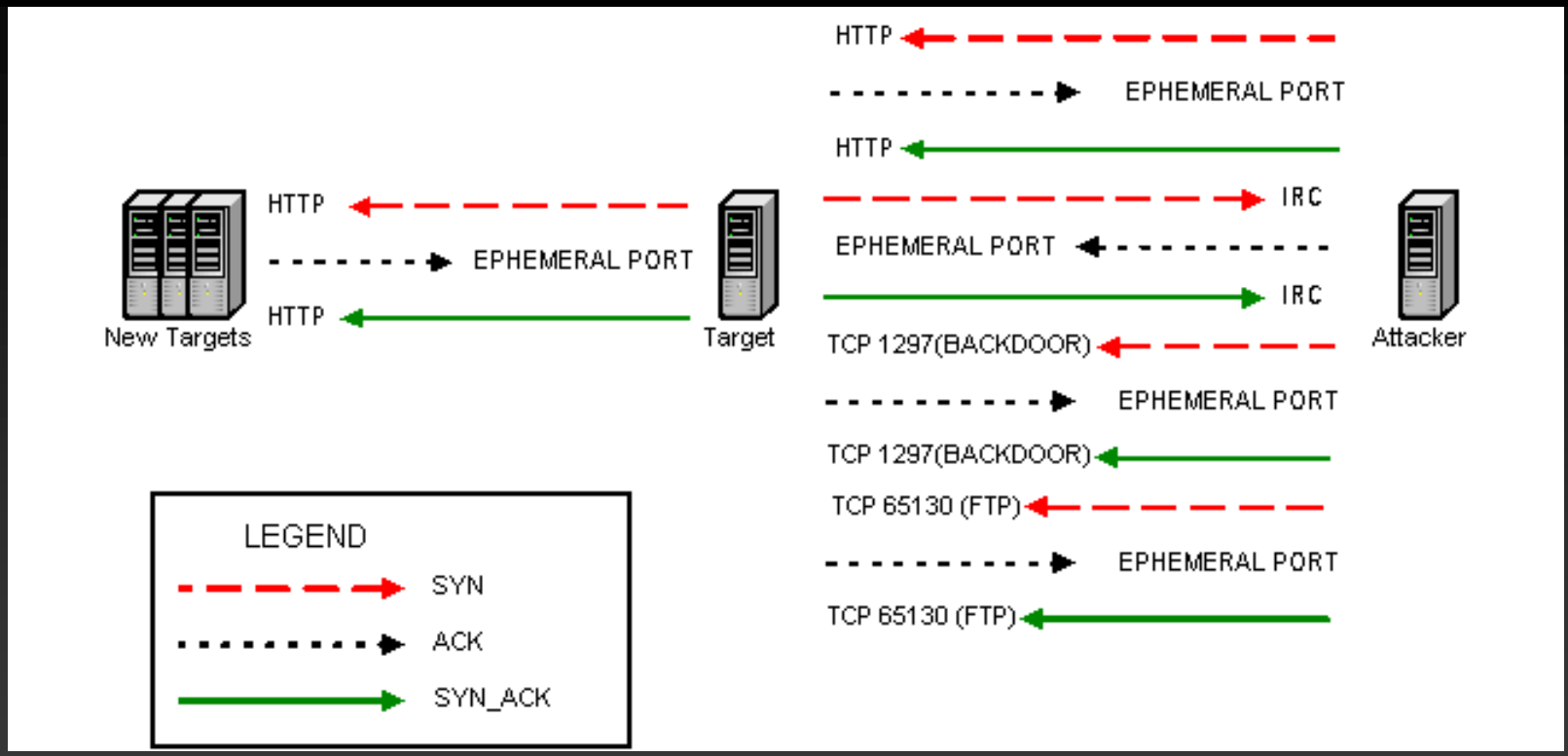
White Paper: Reverse Engineering Malware by Lenny Zeltser

Tkbot: an IRC-using backdoor

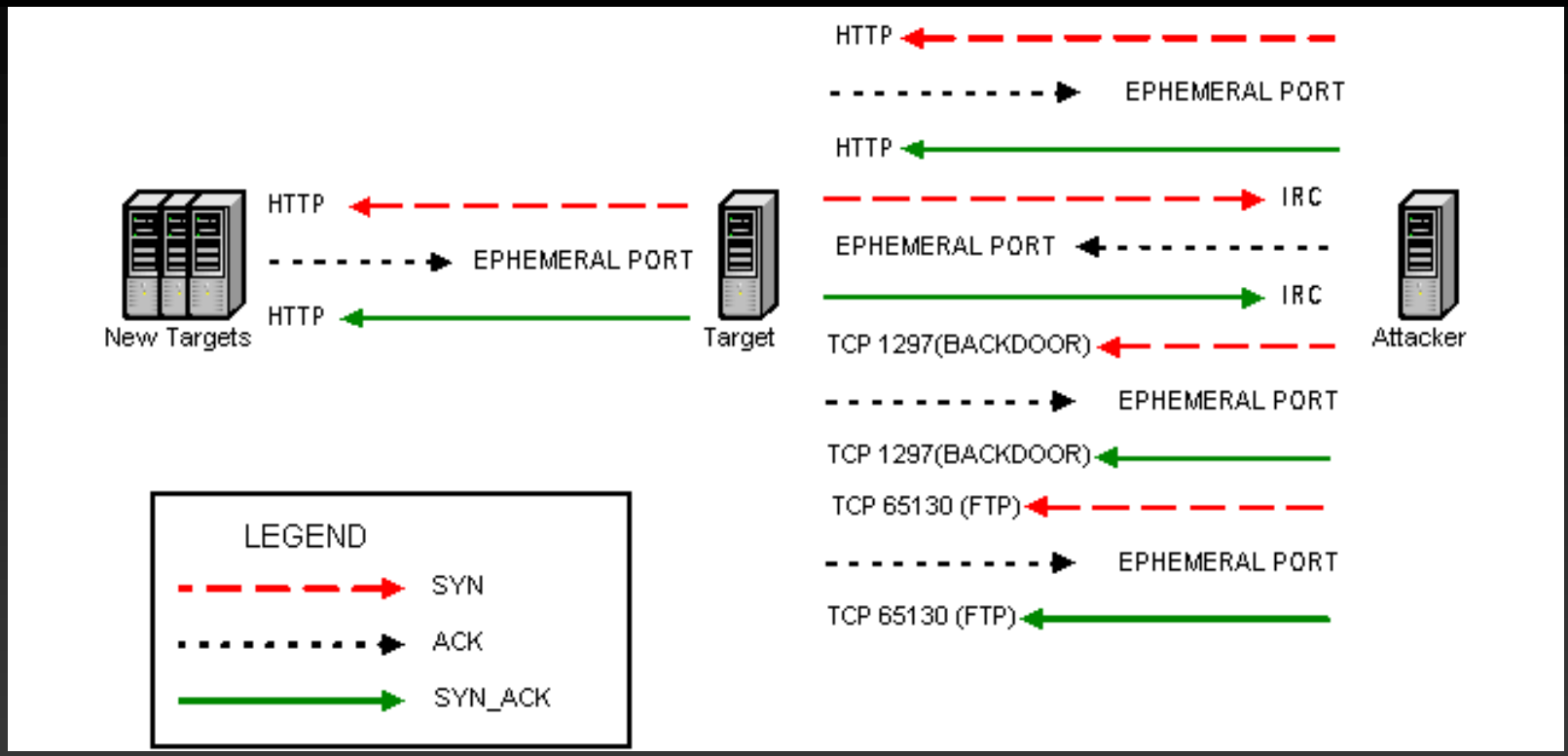


- Masquerades as an OS component
- Installed via IIS exploits
- Uses IRC for communication
- Capable of controlling thousands of hosts (bot code shows increasingly impressive quality, scalability).

tkbot traffic profile (solution?)



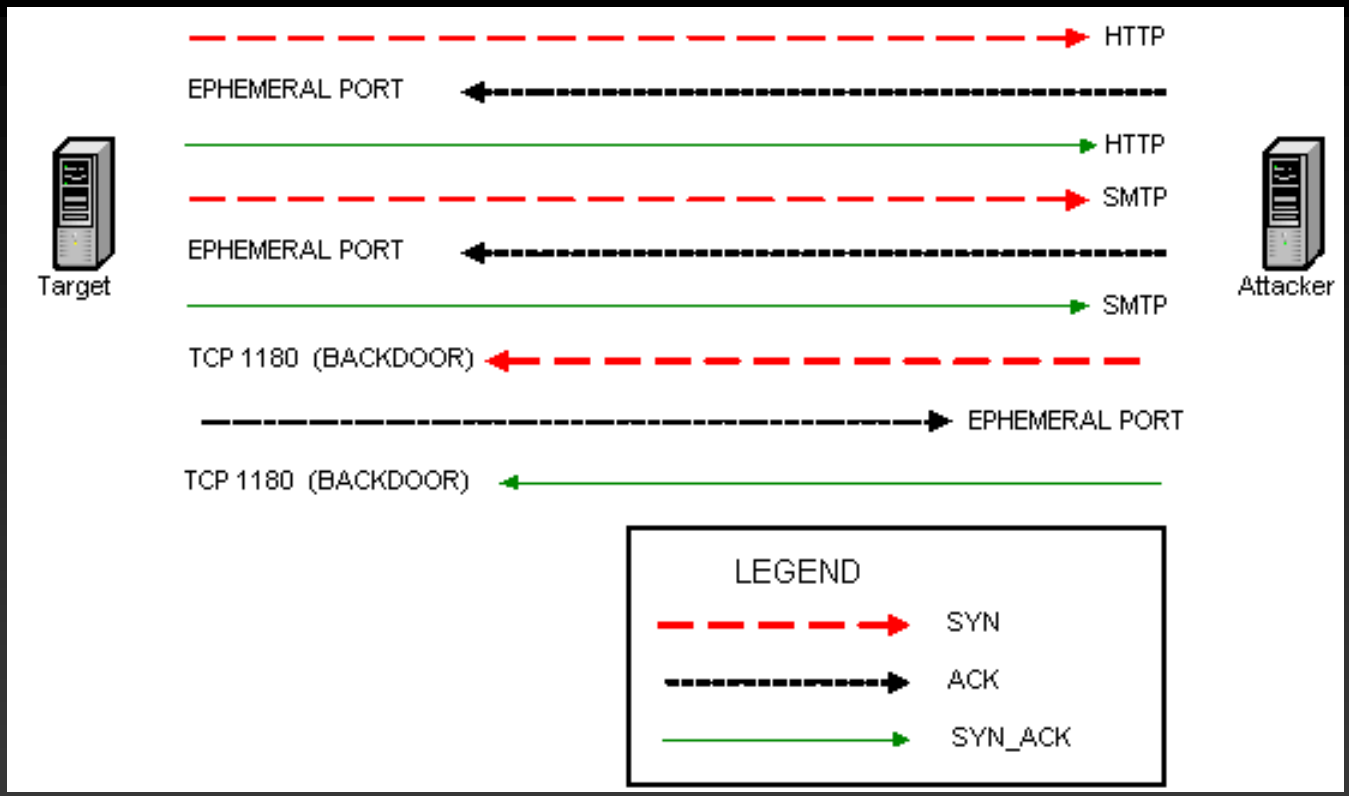
Detection: Outbound IRC.



HTTP Covert Channels

- Can be exploited to gain privileged access / user credentials
- Ports 80 and 443 outbound are the network's Achilles heel
- Asynchronous backdoors can use Port 80 / HTTP outbound for communication
- Very small network footprint; looks like web traffic
- Network devices can authenticate users but not individual processes

HTTP Bot Traffic Profile



HTTP Bot Traffic: how would you detect this?

```
17:46:12.709798 209.58.X.X.1030 > 66.218.X.X.80: S
    2223315203:2223315203(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
17:46:12.795073 66.218.X.X.80 > 209.58.X.X.1030: S 8671164:8671164(0) ack
    2223315204 win 65535 <mss 1460>
17:46:12.795757 209.58.X.X.1030 > 66.218.X.X.80: . ack 1 win 17520 (DF)
17:46:12.836430 209.58.X.X.1030 > 66.218.X.X.80: P 1:142(141) ack 1 win
    17520 (DF)
17:46:12.947188 66.218.X.X.80 > 209.58.X.X.1030: P 1:941(940) ack 142 win
    65535 (DF)
17:46:12.947746 66.218.X.X.80 > 209.58.X.X.1030: F 941:941(0) ack 142 win
    65535 (DF)
17:46:12.947819 209.58.X.X.1030 > 66.218.X.X.80: . ack 942 win 16580 (DF)
17:46:12.976535 209.58.X.X.1030 > 66.218.X.X.80: F 142:142(0) ack 942 win
    16580 (DF)
17:46:13.061977 66.218.X.X.80 > 209.58.X.X.1030: . ack 143 win 65535 (DF)
```

HTTP Bots

- See “A Tale of Two Bots” for more consideration of this in depth
- Includes a case study of an bot using HTTP as a covert channel

HTTP Bots

What is needed is a last line of defense at the desktop layer to enforce policy and stop information leaks when perimeter defenses have been breached.

Digital Guardian

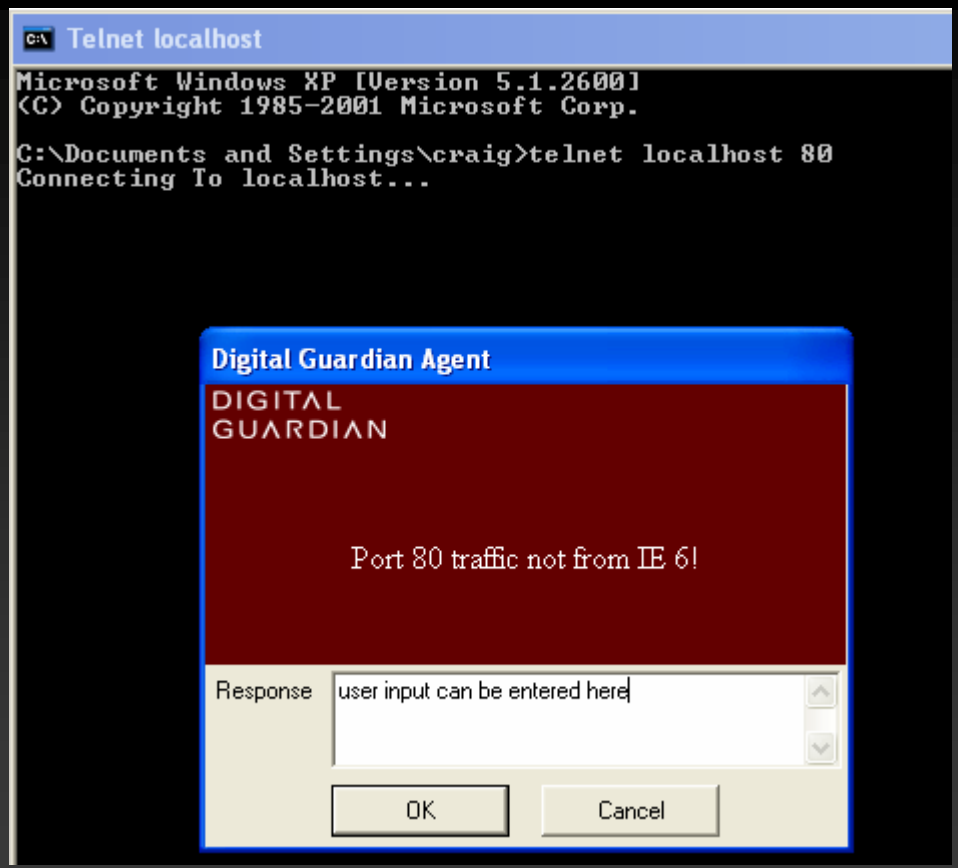
- Users kernel filter drivers to intercept I/O requests and apply policy at the desktop level
- Authentication of processes using MD5 hash
- Centralized management and reporting

Blocking Asynchronous Port-80 using backdoors

IF outbound port IS 80 AND process is Internet Explorer ,allow ; otherwise deny:

```
<and>
<equal property ="netRemotePort" intValue="80"/>
<not>
<!-- [iexplore.exe] -->
<equal property ="opProcessNameHash"
  fileId="QY0wHDsfqUsZWErus9ZRZg=="/>
</not>
</and>
```

Demo



Options:
Prompt user

Alert someone

Block

Unsolved Issues

- Ineffective against a hijacked browser
- Future rootkits implementing separate NDIS layers – bypass TDI layer

20% Insider access

Digital Guardian

- Independent of user privilege and / or access level; effective against stolen / abused passwords
- Enforce policy on a compromised system
- Effective in cases of abused access (e.g. no file copies to removable media)

Q & A