Network Nightmare

Ruling the nightlife between
shutdown and boot with pxesploit

#whoami

- Matt Weeks
- Scriptjunkie if you hang out on irc
- I have a twitter but I don’t use it
- http://www.scriptjunkie.us/
- scriptjunkie {shift+2} scriptjunkie.us

What's going on here

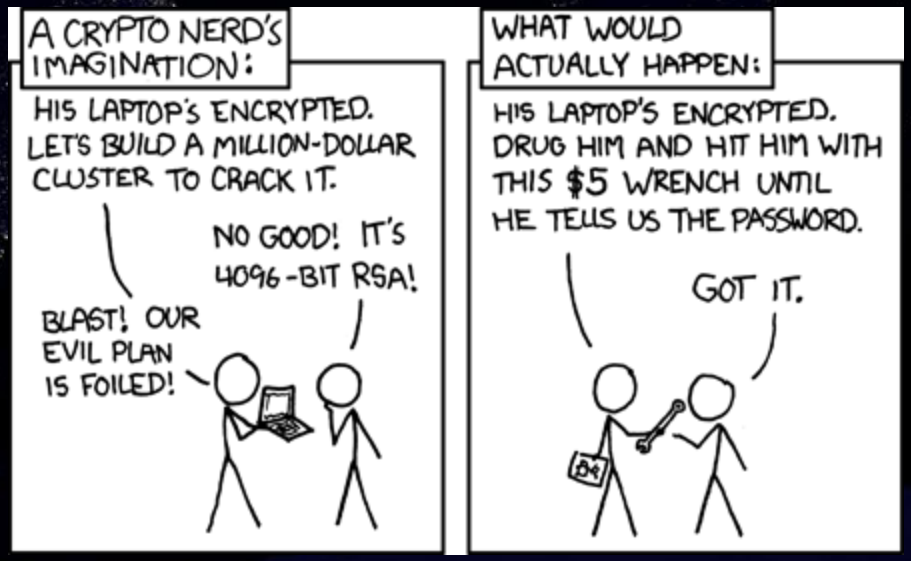
- Want to compromise another system on the LAN?
- Could write an amazing 0-day for [list running network services]
  - But that can take a lot of time
  - Fuzzing/static analysis -> Vulnerability ID -> Identify exploitation path -> Bypass protections -> blah blah blah -> and you still need to escalate privs

Easier way?

- How about we try an offline attack?

Offline attacks

- Evil maid attack
- Rubber hose cryptanalysis



Downsides

- Usually require physical access
- Usually not very stealthy
- Often could wind up with a lot of jail time
- Of course lots of pentesters have flown places, snuck in buildings, and physically accessed systems

PXE

- Intel-introduced firmware to boot from NIC
- BIOS-level access
  - Bypasses application defenses/host firewalls/OS protections/AV
  - Independent of OS
  - Works over network
  - Full system control

How it works

- Step 1 – Your computer shuts down

How it works

- Step 2 – Wake up … something's different

PXE Proliferation

- Almost every system BIOS I have looked at is PXE-capable
- I have no stats on how widely it is turned on
- I have seen it used, I have seen it left on, I have seen it turned off
- I do not have a lot of experience

Why would Intel do this to us?

- My guess at top syadmin reasons:
  - Used for image deployment
  - Used for system restoration
  - Not used, but ready for OS upgrades
  - What's that? I have that on?

How PXE works

- DHCP extension
  - Client sends DHCPDISCOVER with PXE option
  - Server sends DHCPOFFER with server IP addresses, other information
  - Repeat with DHCPREQUEST/DHCPACK
- TFTP Download from identified server
- Executes code
- Magic

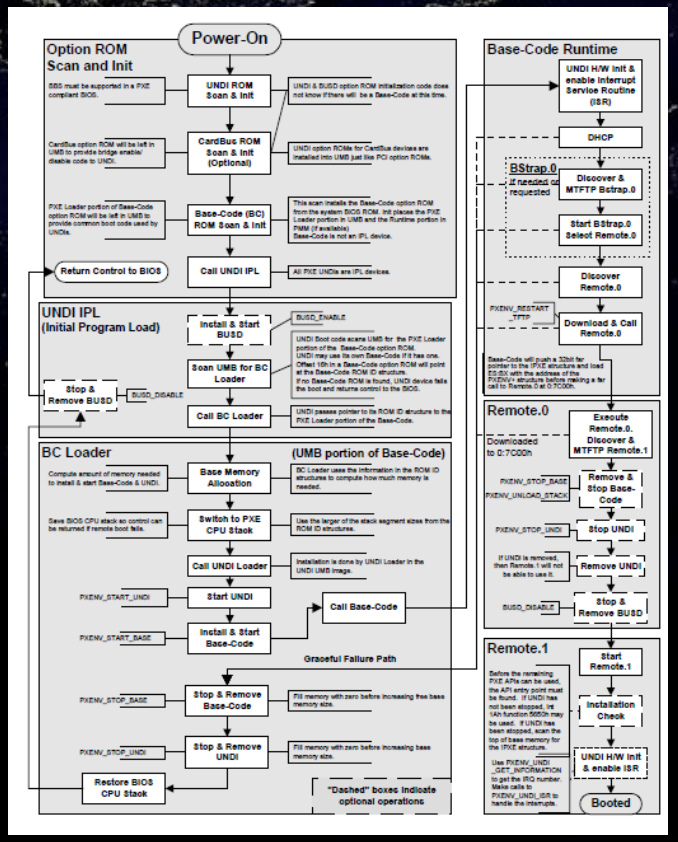
PXE Difficulties

- DHCP extension
  - Must be on LAN, beat real DHCP server
- Forwards to TFTP
  - Need one o' these servers too
- Downloads/executes code
  - Code running on bare metal

PXE Difficulties

Preboot Execution Environment (PXE) Specification

Version 2.1



Current PXE “attacks”

- Manual creation of PXE server
- Manual configuration of DHCP
- Deploying images
- Or running pxelinux

Current PXE “attacks”

- Not written to be attacks
- Manually reconfiguring admin tools
  - Time-consuming
  - Imaging can replace all existing data
  - Difficult to deploy to remote network
  - Unreliable or lack targets
  - Lack support for custom payloads

Online Control

- Some Linux live CDs can be booted via PXE
  - DSL
  - Tiny Core
  - Knoppix
- Strategy
  - Remaster live CD
  - Boot live CD via PXE
    - pxelinux loads kernel, initrd
    - scripts may connect back to nfs to continue booting
  - Have scripts auto-run to connect back
  - Shell!

Online Control

- Demo

Online Control

- Advantages
  - No reliance on target OS
  - Flexibility
  - No need to code the whole attack beforehand

Online Control

- Problems
- MyNetworkCard™ compatibility
  - Even if the distro has a driver for your card, the initrd doesn't!
- Time
  - Someone's probably sitting on the other end staring at the screen
  - Be fast

Offline Code Injection

- You are going to do it anyway
- Executing outside the OS is OK, executing a process with privileges inside the system is better

Offline Linux Code Injection

- Shellcode on boot
  - Write/edit file to RCE
    - /etc/init.d/…
    - ~/.bashrc etc
- User add
  - /etc/passwd
  - ~/.ssh/authorized_keys

Offline Windows Code Injection

- Bootkits
- Binary planting
- Binary swapping
- Binary embedding/modification
- DLL preloading
- Registry edits
- Binary swapping + service editing

Note!

- This presentation will not be addressing FDE
- See cold boot attack or evil maid attack details

Bootkits

- Sinowal
- Stoned
- Whistler
- TDL/Alureon

Bootkits

- Advantages:
  - Skillz points
  - Stealth
  - Full privileges

Bootkits

- Disadvantages:
  - Usually very OS-specific
  - Usually don’t work when MS patches OS protections
  - A lot of work and probably overkill for PXE attack

Binary Planting

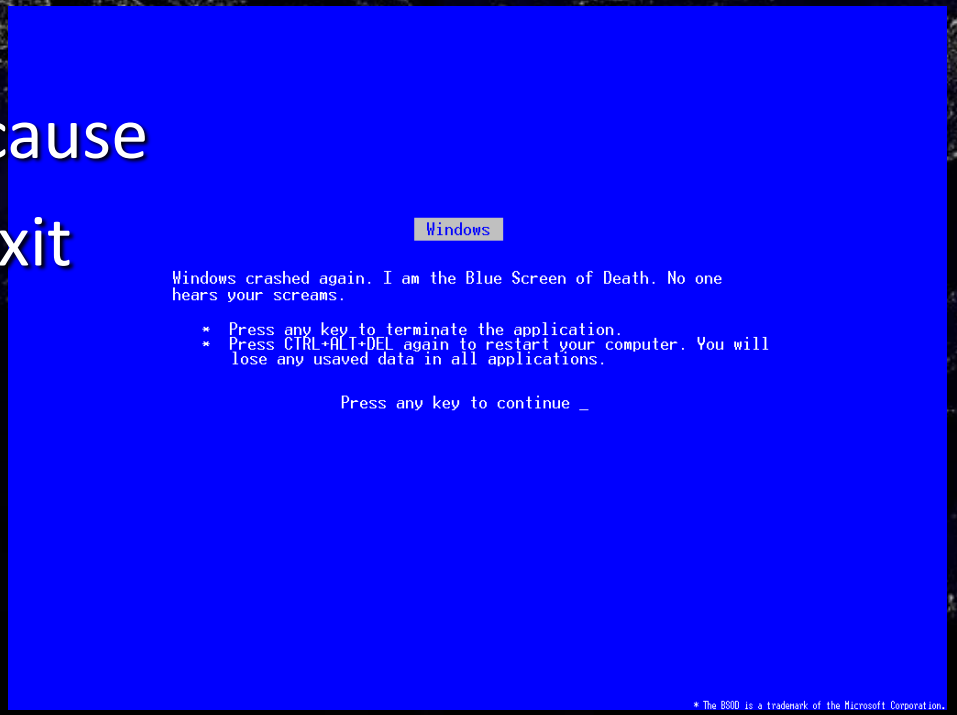
- Startup folders
  - C:\Documents and Settings\All Users\Start Menu\Programs\Startup
  - C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup
  - Unprivileged
- WBEM .mof method
  - **Stuxnet!**
  - Unfortunately not always applicable (Not compatible with Vista+)

Binary Swapping

- Example:
  - Swap services/svchost/wininit/… with replacement
  - Replacement starts up old services.exe and payload, then replaces itself with old services.exe
- Advantages:
  - Code execution guaranteed
  - Privileged
  - Portable

Binary Swapping

- Disadvantages:
  - Early-start processes cause bluescreen when they exit
  - To replace swapped exe, process must exit
  - Later-start processes can be disabled
  - Cannot rely on either



Binary Embedding/Modification

- Inject additional code into existing .exe files
  - svchost/wininit/winlogon/…
- Example:

  msfvenom -f exe -x svchost.exe -k -p - < pay > a.exe

Binary Embedding/Modification

- Problems
- Different architectures
  - Embedding x86 != embedding x64
  - Cannot rely on enough slack space in different windows versions
  - Still have issues with cleaning up after yourself

DLL Preloading

- Swap user32.dll or some other dll
- Or add dll higher in search path with payload
- Problems:
  - Architecture
  - Imports
- Still an option

Registry Edits

- Lots of options!
  - Run keys - HK(LM|CU)\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
    - Reliable
    - Unprivileged
  - Service addition HKLM\SYSTEM\CurrentControlSet\Services
    - Privileged!
    - Registry values differ between versions

Registry Edits

- Service Editing HKLM\SYSTEM\CurrentControlSet\Services
  - Privileged!
  - Changing binpath string, possibly type, start
- Known DLL's
  - Privileged!
  - Add string
- And others

Registry Edits

- Wait, registry edits? Strings?
  - We are using a Linux initrd
  - We are adding data to registry
  - Probably will work with chntpw's ntreged library
  - But …

```
Hive expansion! ... If expansion occured, you will get a
warning when writing back.
```

  - We really don't want to corrupt the HKLM registry, however unlikely

Binary Swapping + Regedit

- Swap a non-essential service binary (late-boot)
- Use DWORD registry edit to enable service
- On boot, service runs

Binary Swapping + Regedit

- Reliable
- No bluescreens
- Cross-arch
- No registry corruption warnings

Pivoting

- Run in memory via meterpreter
  - Railgun
    - Network delay
  - Extension
    - Compiled program

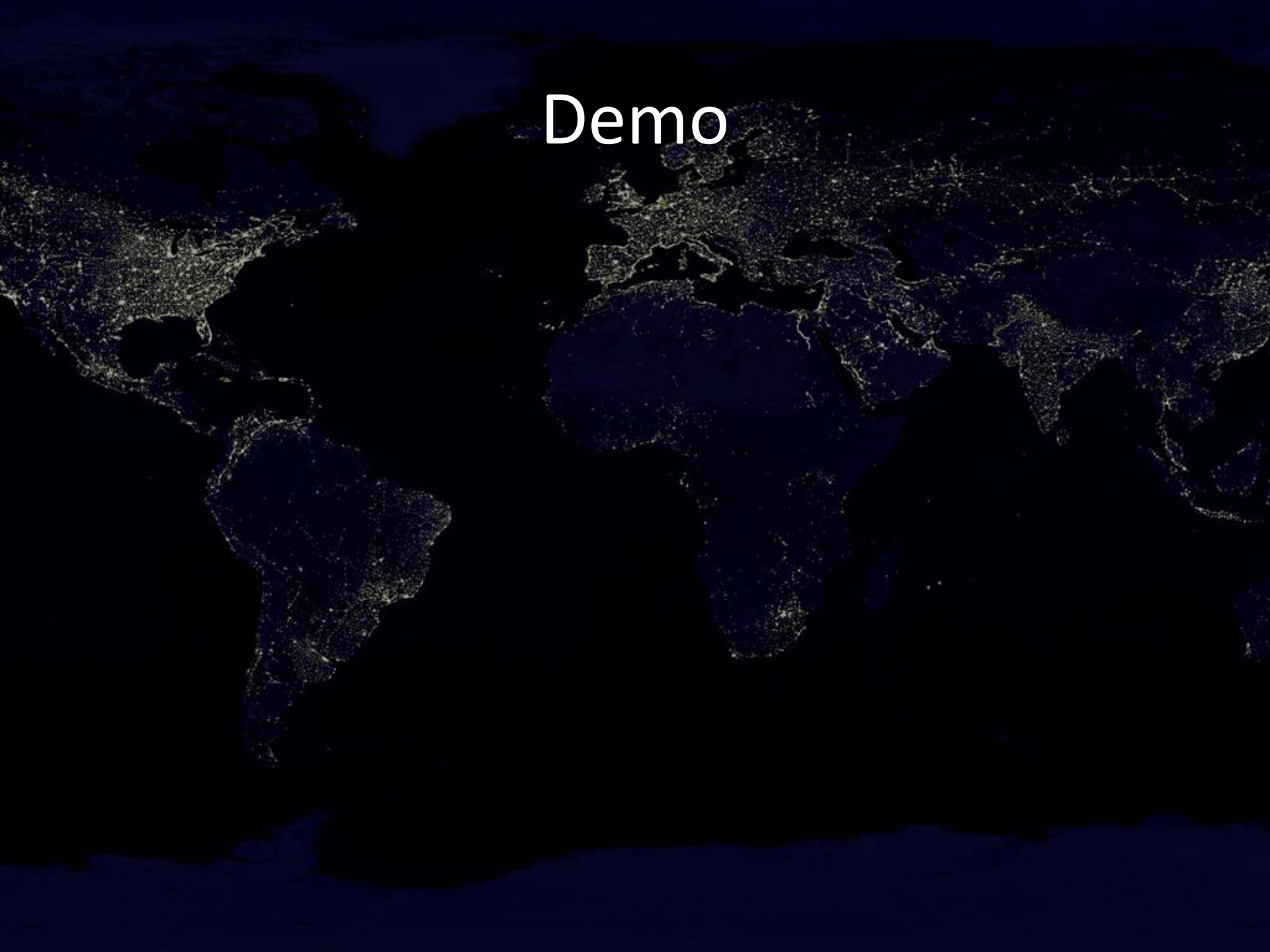
Meterpreter Review

- TLV request
- Embedded DLL
- Reflective Loader
- Method Calls

Attack Recap

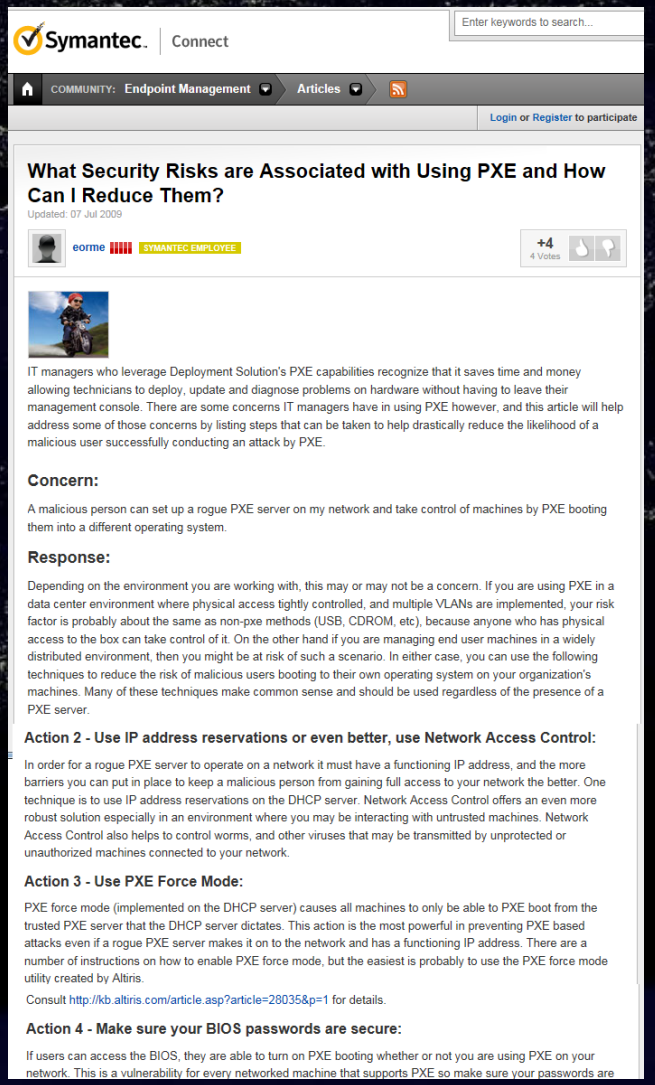
1. Dynamic payload generation
2. DHCP forwarder
3. TFTP serve
4. PXELinux kernel, initrd load
5. Binary swap
6. Registry edit
7. Reboot to OS
8. Swapped EXE spawns payload, cleanup

Demo



Defense

- How to fail at defense:
  - IP reservations
  - NAC
  - PXE Force Mode
  - BIOS passwords



Symantec. | Connect

Enter keywords to search...

COMMUNITY: Endpoint Management | Articles

Login or Register to participate

What Security Risks are Associated with Using PXE and How Can I Reduce Them?

Updated: 07 Jul 2009

eorme SYMANTEC EMPLOYEE

+4 4 Votes

IT managers who leverage Deployment Solution's PXE capabilities recognize that it saves time and money allowing technicians to deploy, update and diagnose problems on hardware without having to leave their management console. There are some concerns IT managers have in using PXE however, and this article will help address some of those concerns by listing steps that can be taken to help drastically reduce the likelihood of a malicious user successfully conducting an attack by PXE.

Concern:

A malicious person can set up a rogue PXE server on my network and take control of machines by PXE booting them into a different operating system.

Response:

Depending on the environment you are working with, this may or may not be a concern. If you are using PXE in a data center environment where physical access tightly controlled, and multiple VLANs are implemented, your risk factor is probably about the same as non-pxe methods (USB, CDROM, etc), because anyone who has physical access to the box can take control of it. On the other hand if you are managing end user machines in a widely distributed environment, then you might be at risk of such a scenario. In either case, you can use the following techniques to reduce the risk of malicious users booting to their own operating system on your organization's machines. Many of these techniques make common sense and should be used regardless of the presence of a PXE server.

Action 2 - Use IP address reservations or even better, use Network Access Control:

In order for a rogue PXE server to operate on a network it must have a functioning IP address, and the more barriers you can put in place to keep a malicious person from gaining full access to your network the better. One technique is to use IP address reservations on the DHCP server. Network Access Control offers an even more robust solution especially in an environment where you may be interacting with untrusted machines. Network Access Control also helps to control worms, and other viruses that may be transmitted by unprotected or unauthorized machines connected to your network.

Action 3 - Use PXE Force Mode:

PXE force mode (implemented on the DHCP server) causes all machines to only be able to PXE boot from the trusted PXE server that the DHCP server dictates. This action is the most powerful in preventing PXE based attacks even if a rogue PXE server makes it on to the network and has a functioning IP address. There are a number of instructions on how to enable PXE force mode, but the easiest is probably to use the PXE force mode utility created by Altiris.

Consult http://kb.altiris.com/article.asp?article=28035&p=1 for details.

Action 4 - Make sure your BIOS passwords are secure:

If users can access the BIOS, they are able to turn on PXE booting whether or not you are using PXE on your network. This is a vulnerability for every networked machine that supports PXE so make sure your passwords are

Defense

- VLAN isolation
  - Each system on separate VLAN
  - Localize broadcast domains
  - Forward DHCP traffic
  - Configure via enterprise switch/routers

Defense

- Firewalls
  - Only allow DHCP traffic to/from server
  - Watch for ARP poisoning

Defense

- Detection of rogue DHCP servers
  - Scan periodically
  - Check for duplicate replies
  - Check for ARP poisoning
  - Check for unregistered clients if possible

Questions