

Injecting Trojans via Patch Management Software & Other Evil Deeds

Today's Key Topics

- *Patching up close*
- *Anatomy of a patch*
- *The process & the system*
- *Design and implementation flaws*
- *Abusing the system*
- *Other evils deeds*
- *Defending the system*
- *Summary*
- *Q&A*

Background Info

My Background

- Blah, blah, blah...read the bio
 - Fascinated with twisting commercial software
 - Fav tool, toy or talk
 - Cazz' (Shmoo) Snort+Perl+Metasploit

Major Kudos to Steve Manzuik

- Founder/moderator of Vulnwatch
- Co-author "Hack Proofing Your Network" 2nd Ed.

Thanks to Tracy Elpers

Disclaimer

- **Research is still in progress**
 - **Vendors w/ verified flaws will be worked with**
- **No vendor/product & with any specific flaw will be singled out by name today**
 - **(unless already public info)**
- **Just because a vendor is mentioned, doesn't mean they have a problem**
- **Any security flaws discussed today may apply to multiple vendors**
- **Exploit not in the wild (yet)**

Patching Up Close

- **Why patch management?**
 - Improve security & uptime
- **How big is the problem?**
 - Standard corporate servers, workstations, laptops
 - What about handheld devices?
 - What about consumer versions?
 - What about phones, cable set-top box?
 - Media centers, xbox, ???
- **Is this a mission critical app?**
 - Primary remediation tool for many organizations

Patching Up Close

- **Patching should be easy (not)**
 - Extensive patching expertise exists?
 - MSFT has worked to make things easier for us?
 - 2002 – 154 security patches
 - 2003 – 174 security patches
 - 2004 – 172 security patches
 - Few standards for patches
 - Complexity
 - Tools have limited view of config data
 - Scale of enterprises
 - Shift, Drift and Shadow IT

Why Provisioning Isn't Enough

- *Images Rolled Out to the 'Standard'*
 - The 'Standard' changes all the time
 - Patches, performance issues, risk mitigation
- *New Images Take Time to Create and Test*
 - When do they get rolled back out?
- *Many Shops Simply 'Ghost it'*
 - If the machine (running the image) was compromised and you re-imaged
...PERPETUAL SITTING DUCK!!

Host Security Relies on More Than Patching or Provisioning

- *What about?*
 - Password mgmt, Guest Accounts, Registry Settings
 - Spyware, Rogue applications (P2P, IM), Antivirus
 - Web apps, CRM, ERP
- **Patched \neq Secure**

Anatomy of a Microsoft patch

- **Digitally signed binary from MSFT**
 - Extras associated with a patch
 - mssecure.cab (mssecure.xml)
 - Security bulletin
- **3rd party patches**
- *“Patch Tuesday”*
 - **Why once a month?**

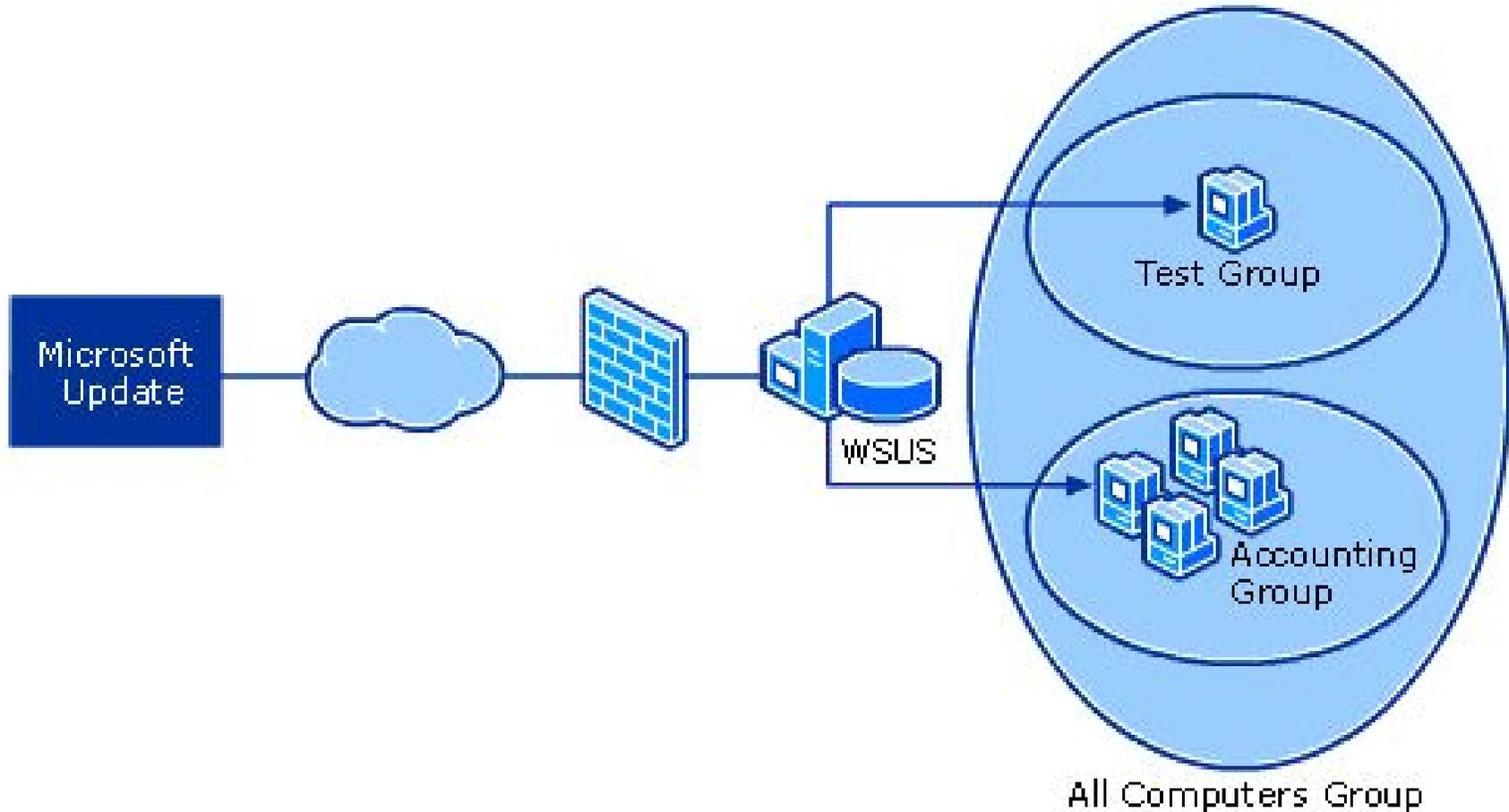
The Process

- **Good scenario (not that common)**
 - Vendor finds bug/get notified about bug
 - Vendor validates, tests and fixes bug
 - Vendor notifies customer & releases patch
 - Customer receives, validates & tests patch
 - Customer rolls out patch in timely manner
 - Customer updates production images
- **Problems with the process?**

The System

- **Types of solutions**
 - Patch management specific
 - Software distribution/systems mgmt tools
- **Platform support**
- **Architectural considerations**
 - Agent vs Agentless
 - Mobile clients
 - Remote distribution sites

The System-WSUS

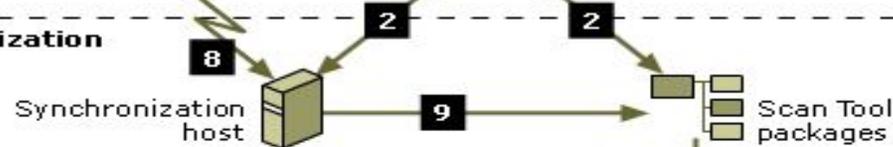


The System-SMS w/FP

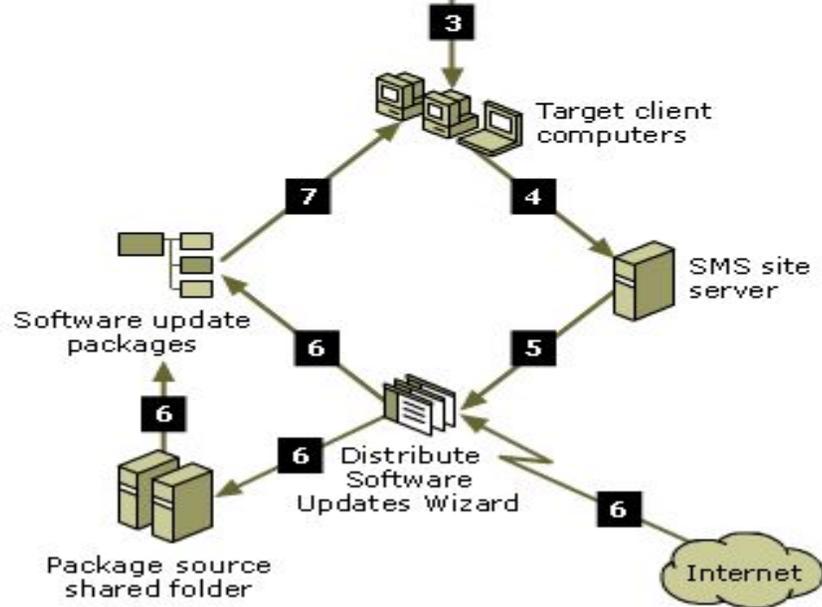
Installation Phase



Synchronization Process



Scan and Update Process



The System

- **Communication**
 - Internal is usually RPC/DCOM (sometimes HTTP)
 - Updates via HTTP
- **Encryption**
- **Authentication**
- **Integrity checking**
- **General issues with the system?**

Other Design & Implementation Flaws

- **Digital signatures**
 - Validation issues from source, at distribution, at target host
- **Patch/packages/repackaging**
 - ACLs and roles are usually weak
 - Custom packaging, repackaging
 - No signature or invalid signature
 - Which patch is that really?

Abusing the System-Scenario #1

- **Internal scenario**

- Compromise the patch repository
 - Sniff the network for credentials
 - Access patches/packages via improper ACLs
 - Compromised package gets distributed
- Mess with patch targeting
- MITM and substitute payload
- Worst case scenario, the system is owned
 - Can be used to cause damage
 - Can't be used for remediation

Abusing the System-Scenario #2

- **External scenario**

- DNS Hijack/Spoof attack
 - In coordination with 'Patch Tuesday' begin redirecting requests looking for source
 - Redirect URLs like windowsupdate.microsoft.com, download.microsoft.com, vendorname.com
- Effective attacker would wait until there is a major issue that a lot of people will want to patch

Abusing the System-Scenario #2

- **The trojan patch (cntd)**
 - Introduce a trojan patch
 - Could actually address a real problem
 - Trojan patch also contains payload of choice
 - Trojan patch can be digitally signed
 - Not with a MS key as obtaining a legitimate MS signing key would be hard
 - Still effective because only a few tools check for a signature, even less check the legitimacy of that signature

Other Evils Deeds

- **DoS the network with packages**
- **DoS the system-agent status issue**
- **Enterprise scalable BSOD**
- **Leverage the system to disable other host security**
- **This just affects Microsoft platforms, right?**

Defending the System

- **Fix the process (not just the product)**
- **Evaluate quarantine solutions**
- **ACLs & roles**
- **Ensure that all packages have valid signatures, at all stages**
- **Keep an eye on network services like DNS**
- **Vendor improvements**

Summary

- **Abuse of Patch Mgmt/System Mgmt tools has potential to take down organizations**
 - Problems exist w/ the process, system and implementation
- **Don't rely too much on patching for security**
- **Organizations should take corrective actions now before exploits appear**
- **Vendors need to make changes**

Questions?

Chris Farrow

Chris.farrow@configuresoft.com

Steve Manzuik

steve.manzuik@configuresoft.com