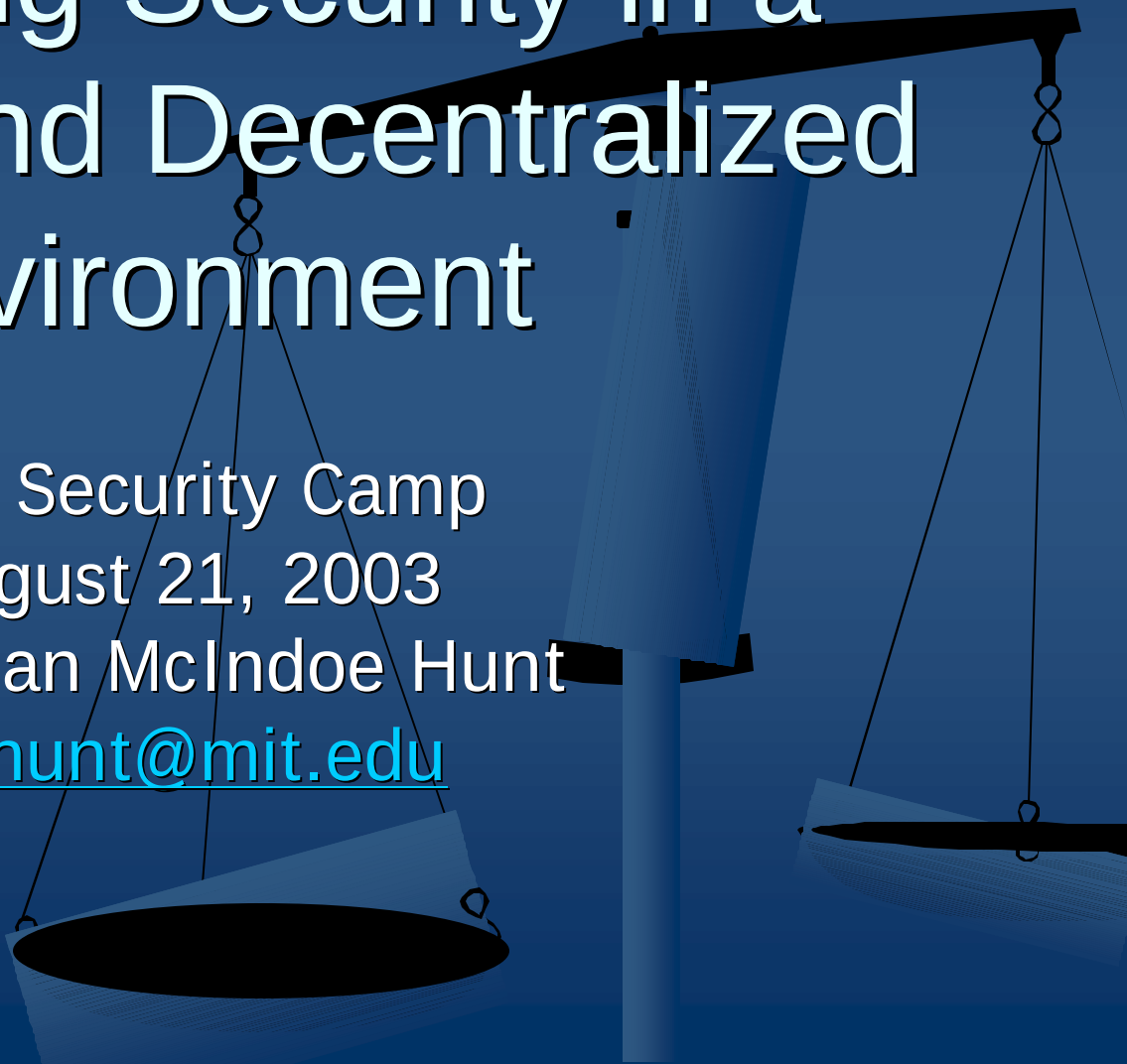
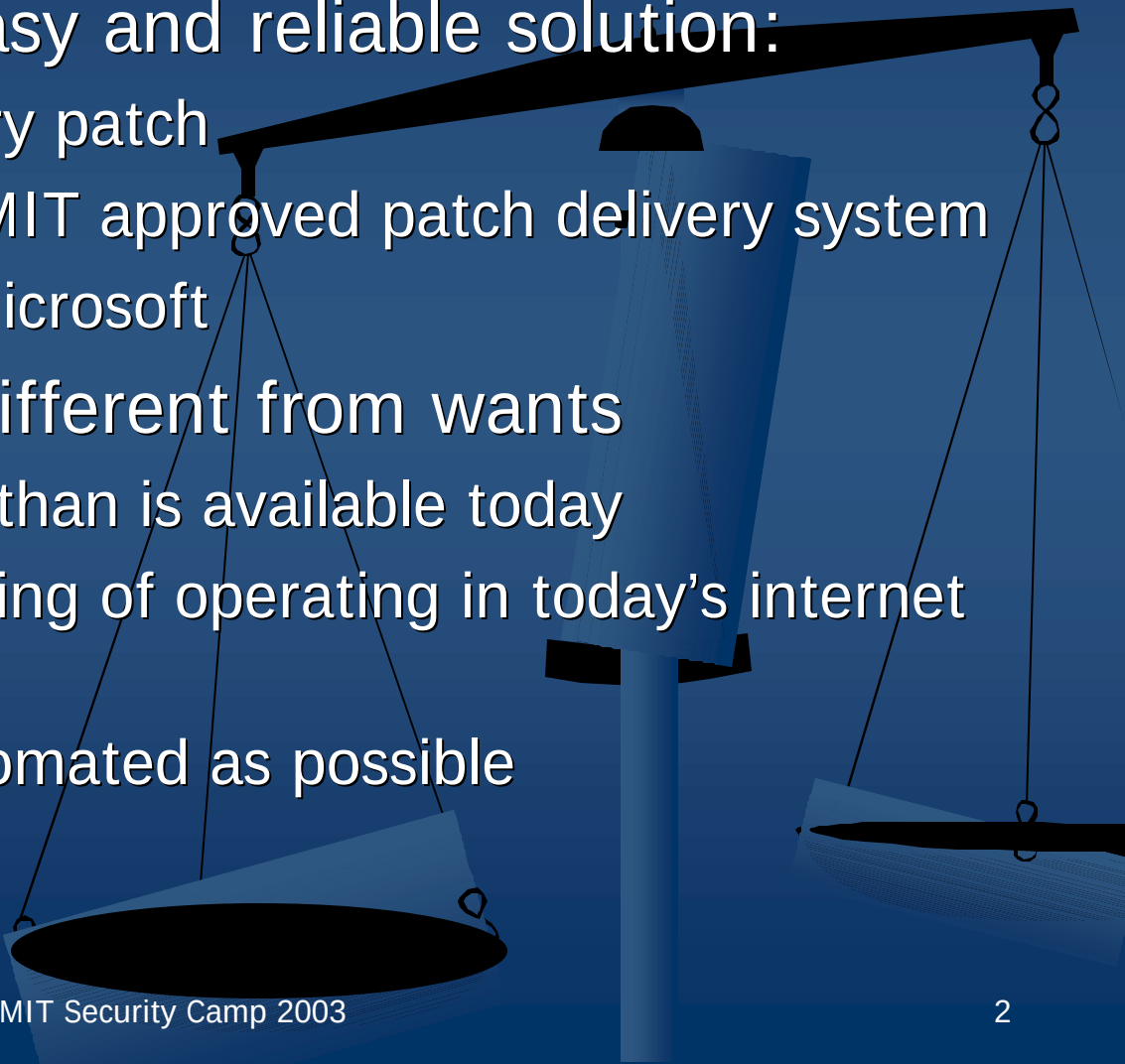
Improving Security in a Diverse and Decentralized Environment



MIT Security Camp
August 21, 2003
Jonathan McIndoe Hunt
jmhunt@mit.edu

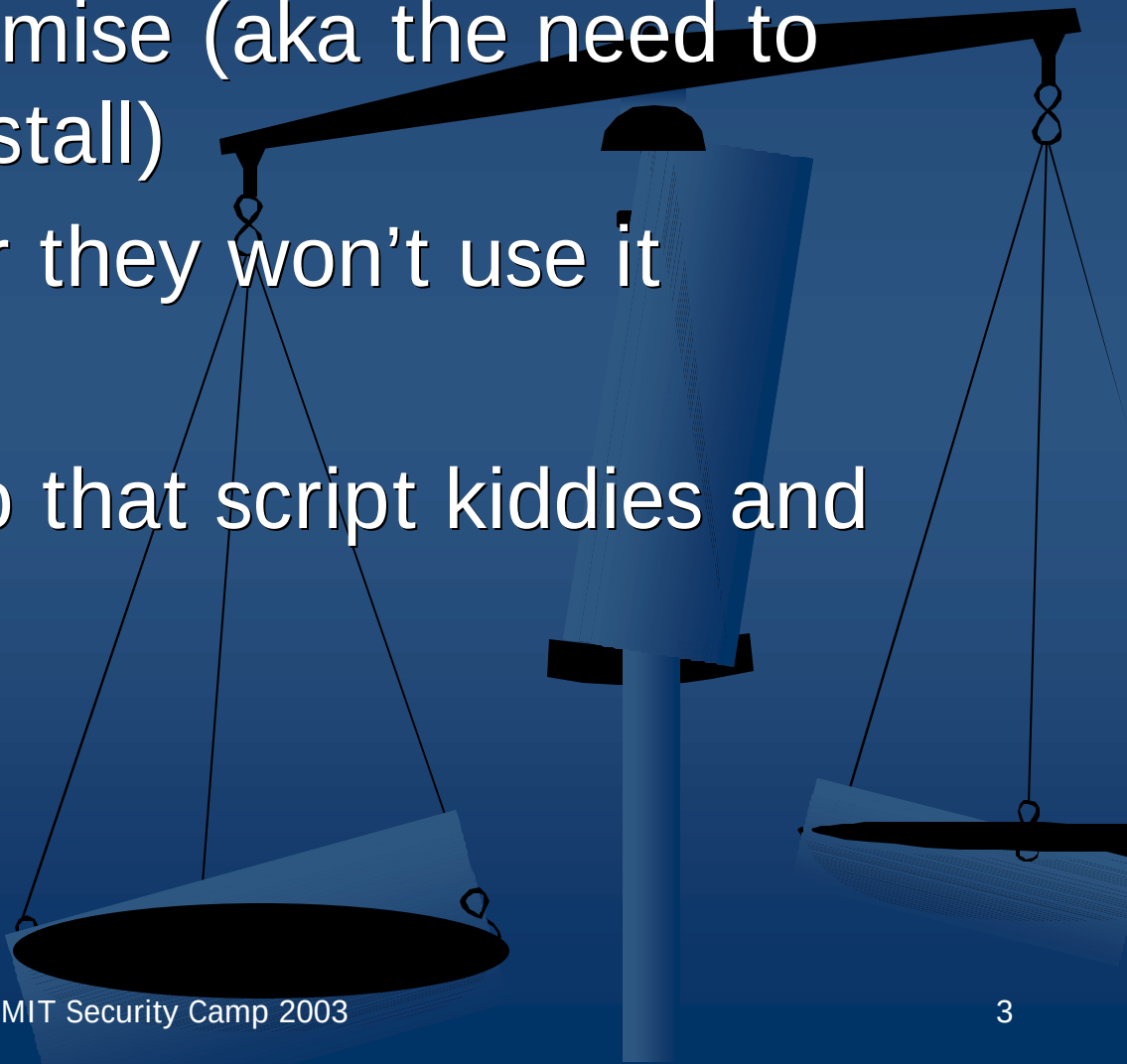
After MS03-026 exploit

- Clients want an easy and reliable solution:
  - IS to validate every patch
  - IS to develop an MIT approved patch delivery system
  - They don't trust Microsoft
- Clients need are different from wants
  - Something easier than is available today
  - Better understanding of operating in today's internet
    - Security Mindset
  - Something as automated as possible



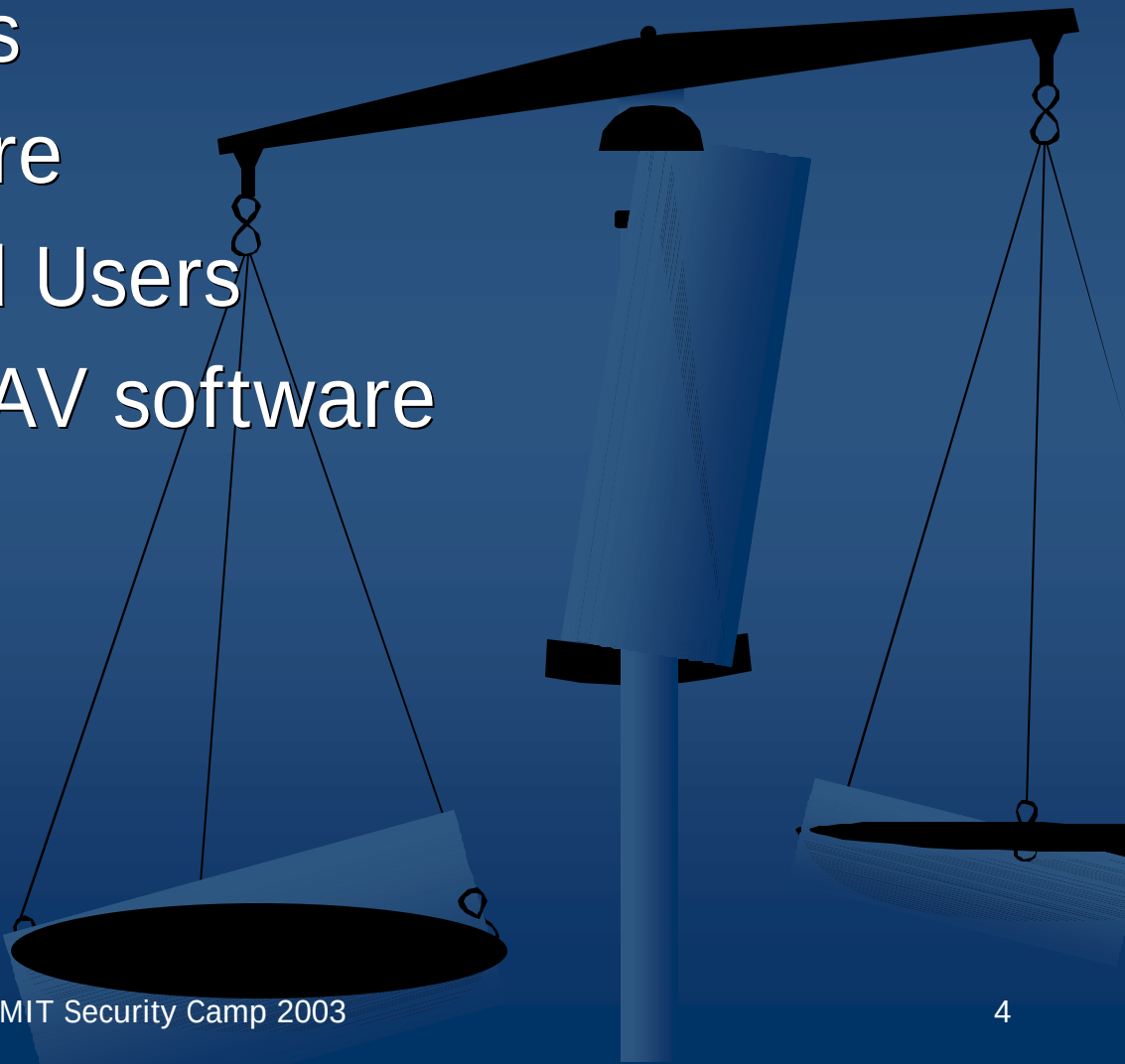
Goals of Security

- Prevent compromise (aka the need to reformat & reinstall)
- Easy to use – or they won't use it
- Raise the bar so that script kiddies and worms skip us



Security

- Good passwords
- Patched Software
- Smart/Educated Users
- Easy tools, like AV software

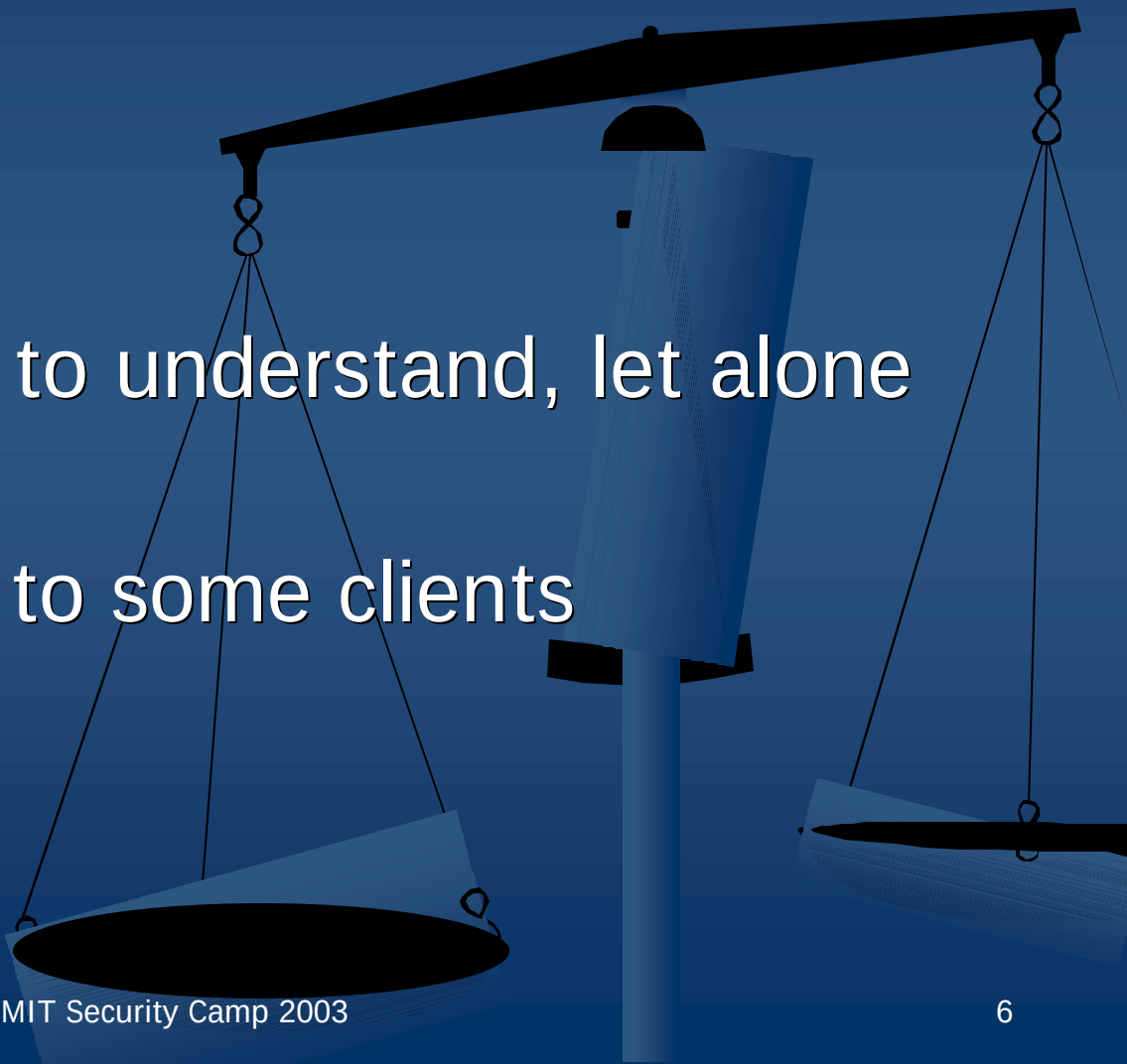


Primary Windows Tools

- Local Security Policy with Good passwords
- Automatic Updates/Windows Update
- Anti-Virus Software

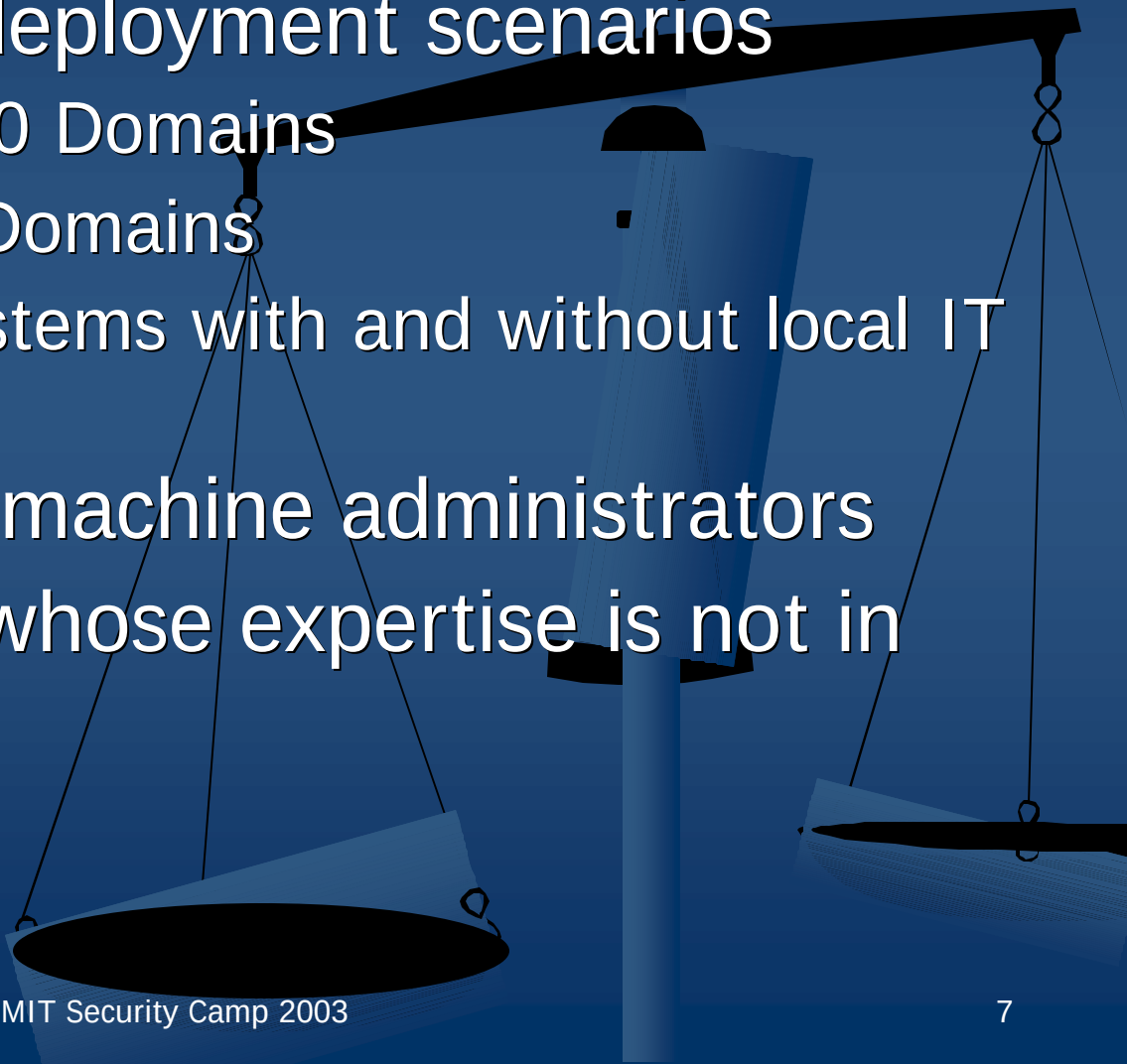
Obstacles

- Environment
  - Diversified
  - Decentralized
- Security is hard to understand, let alone implement
- Need is unclear to some clients



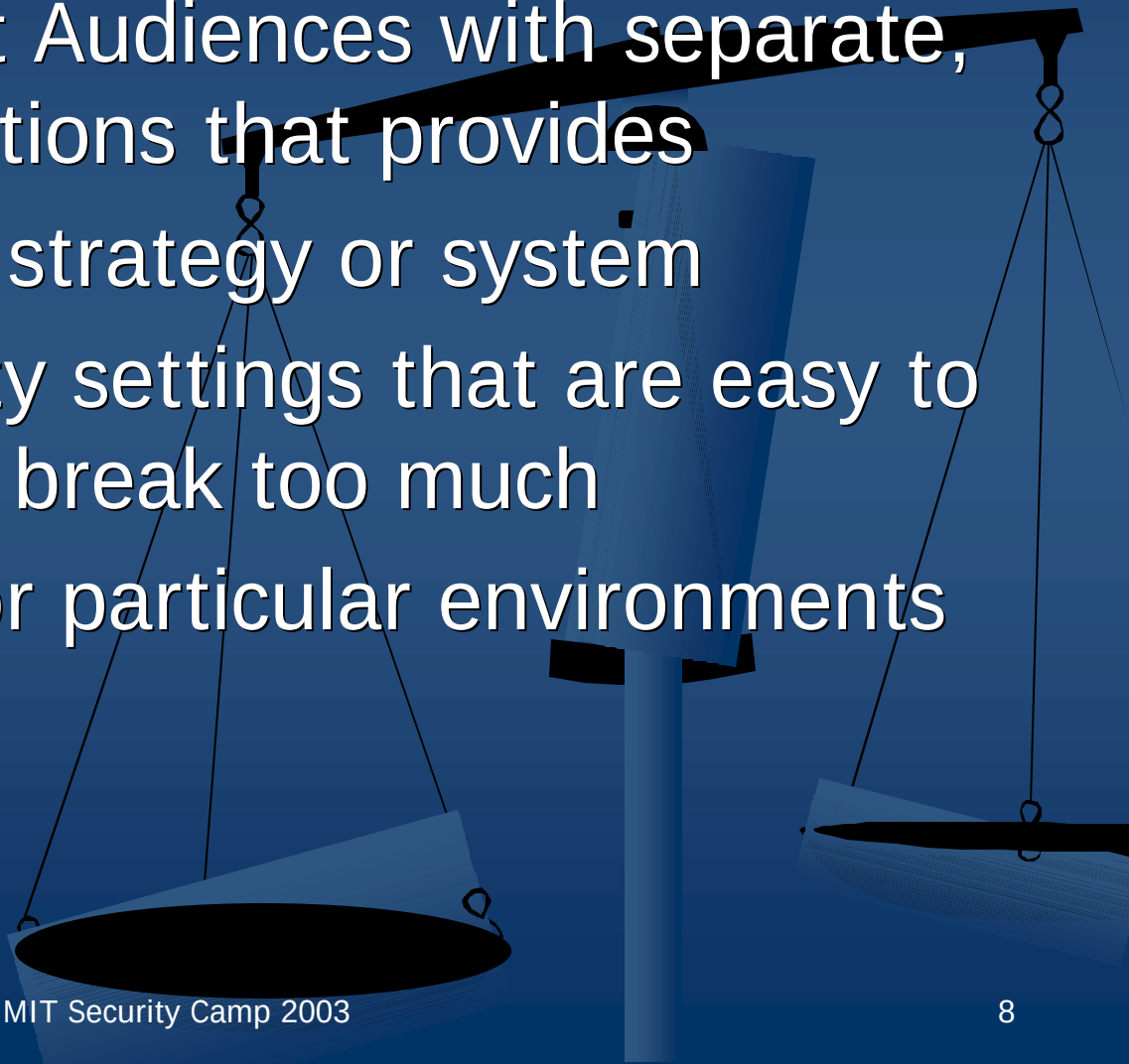
Environment (MIT)

- Wide range of deployment scenarios
  - Managed NT 4.0 Domains
  - Central Win2k Domains
  - Stand Alone systems with and without local IT help
- Knowledgeable machine administrators
- Administrators whose expertise is not in computers
- No firewall



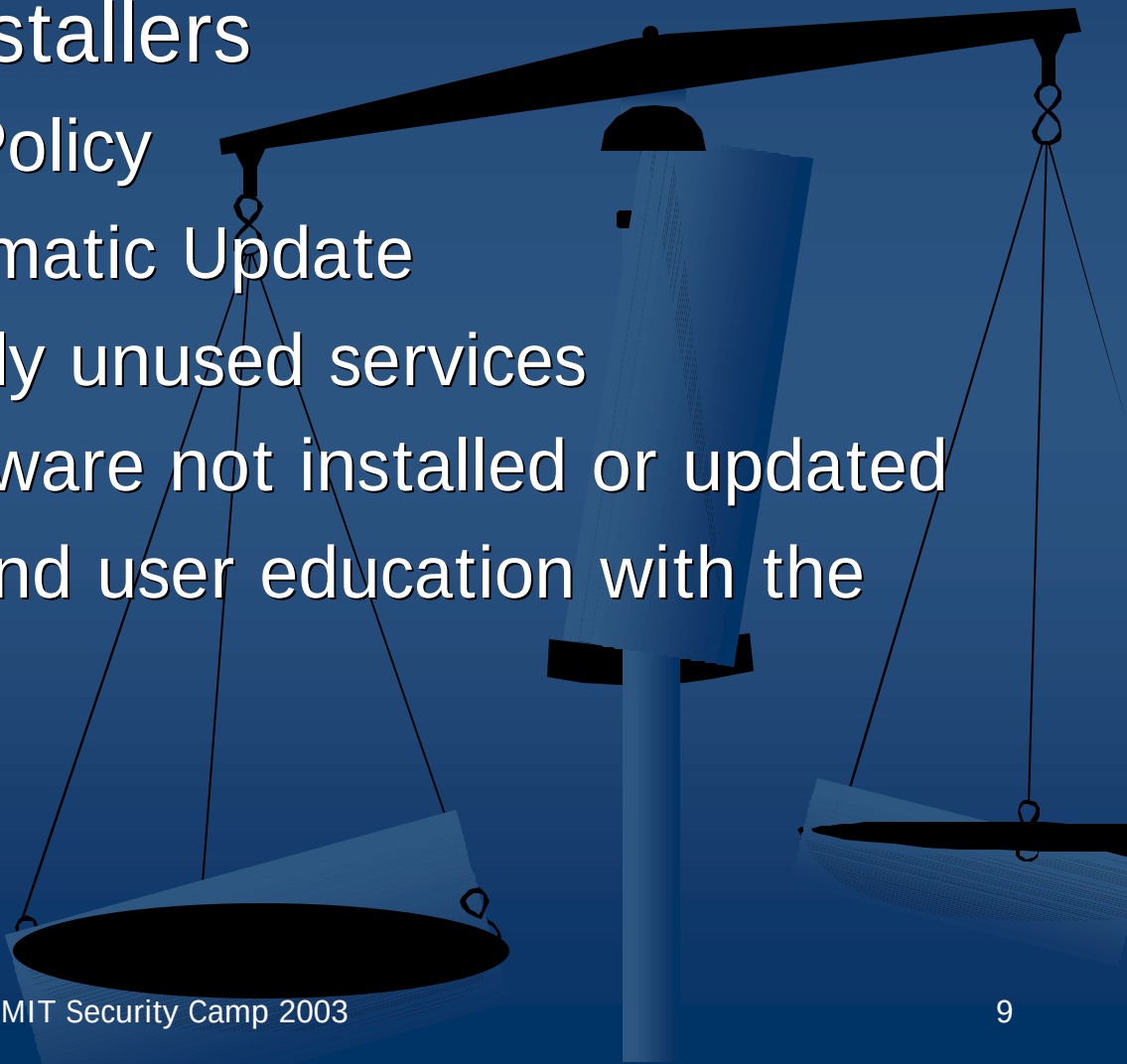
Proposed Solution

- Target Different Audiences with separate, but related solutions that provides
- Routine update strategy or system
- Effective security settings that are easy to apply and don't break too much
- Customizable for particular environments



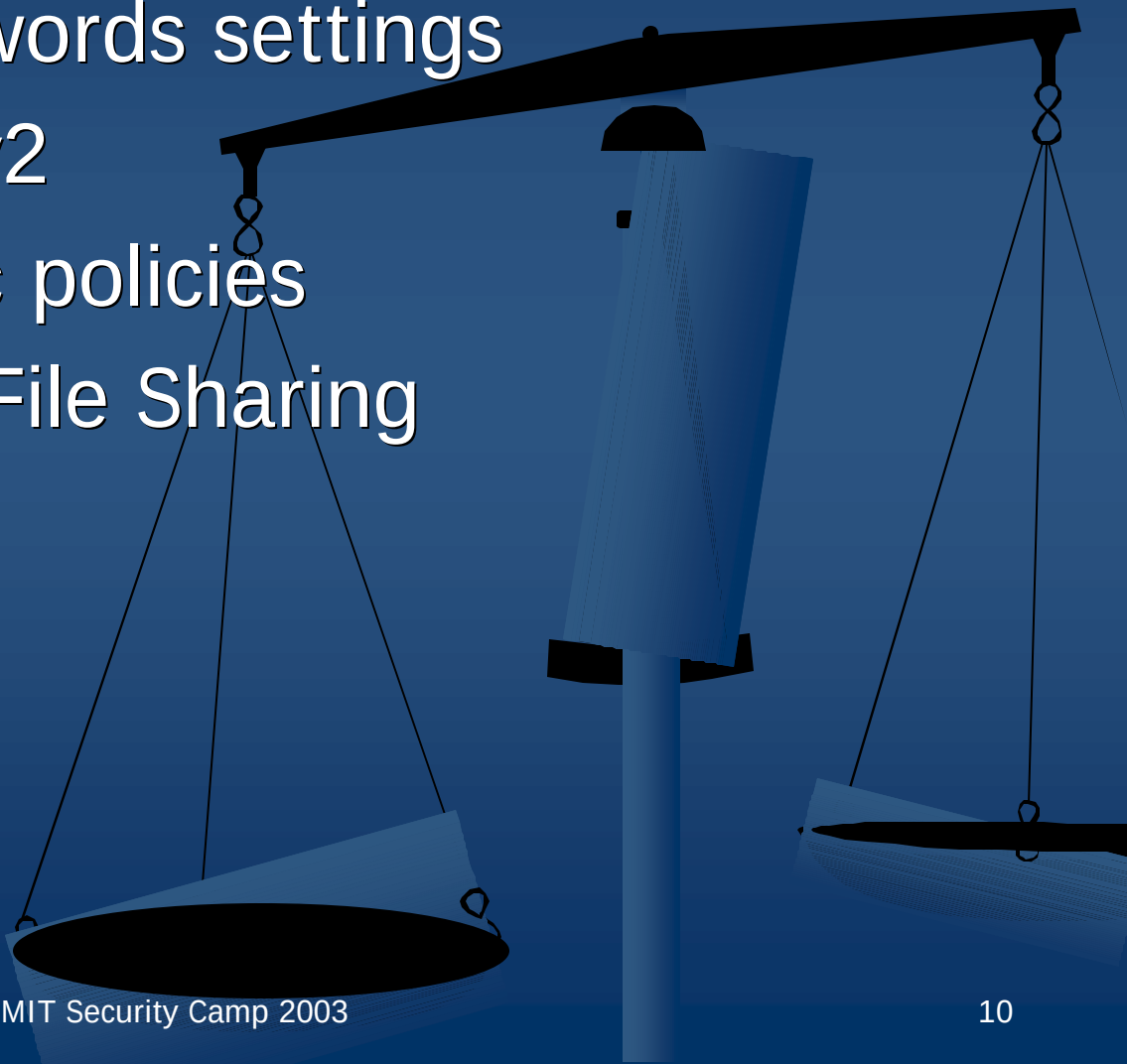
Stand Alone systems

- MIT Security Installers
  - Local Security Policy
  - Configure Automatic Update
  - Disable generally unused services
  - Warn if AV software not installed or updated
  - Provide some end user education with the installer



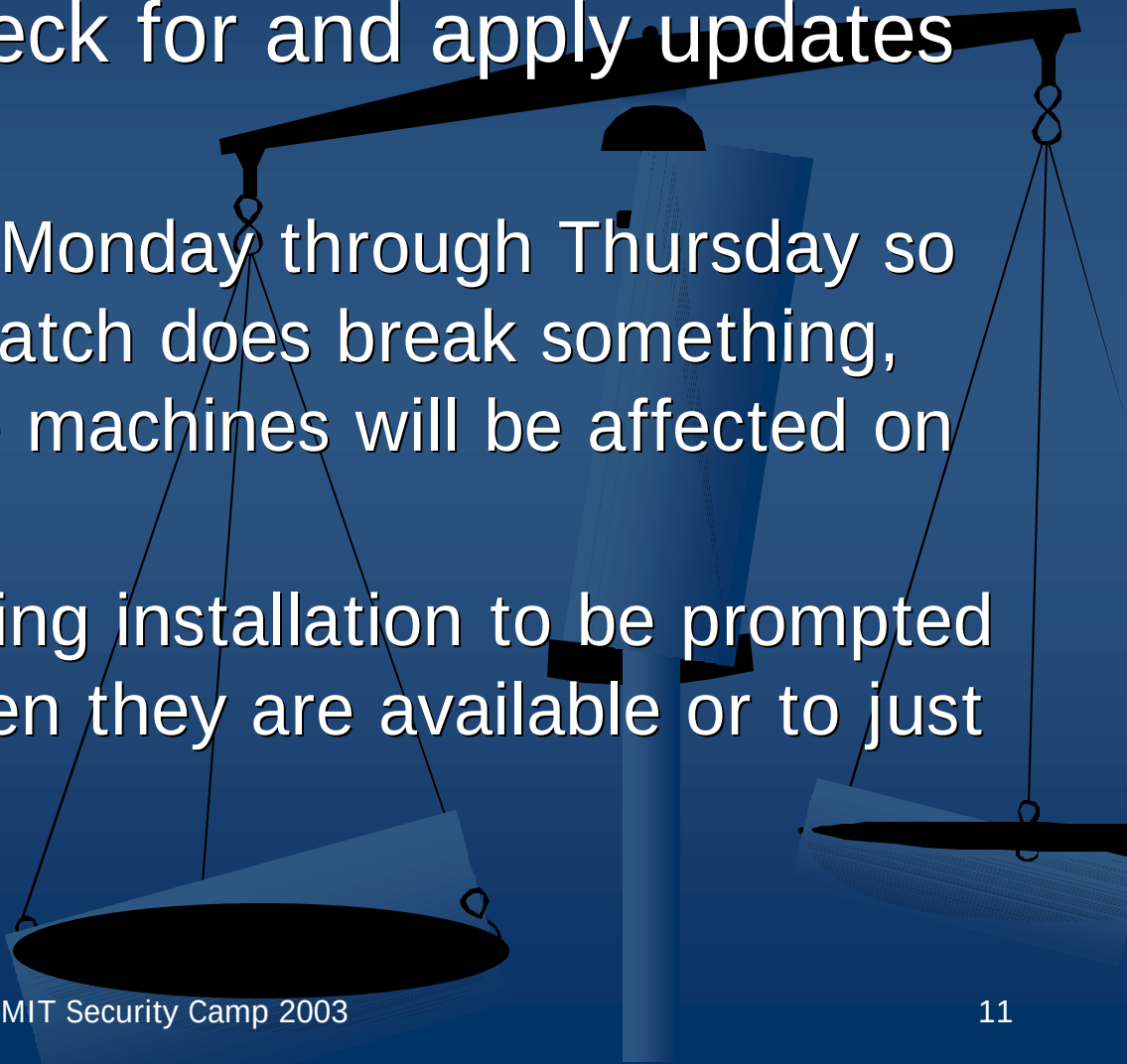
Local Security Policy

- Configure passwords settings
- Require NTLM v2
- Configure IPSec policies
- Disable Simple File Sharing



Configure Automatic Update

- Configure to check for and apply updates weekly
  - RNG 1-4 select Monday through Thursday so that in case a patch does break something, only a ¼ of the machines will be affected on a given day
  - Give option during installation to be prompted for updates when they are available or to just install them

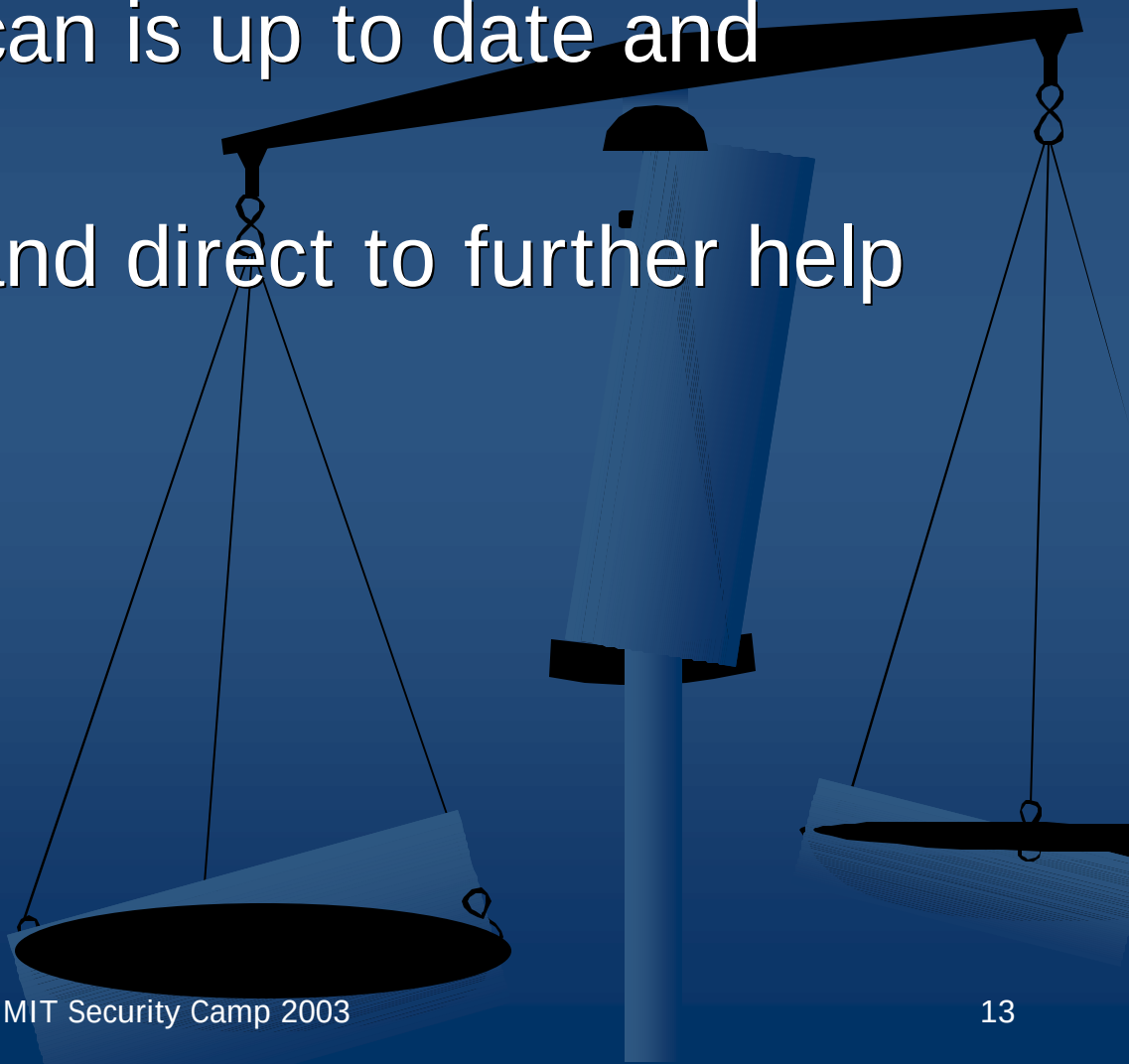


Disable unused services

- Windows Messenger Service
- Others to be determined & debated

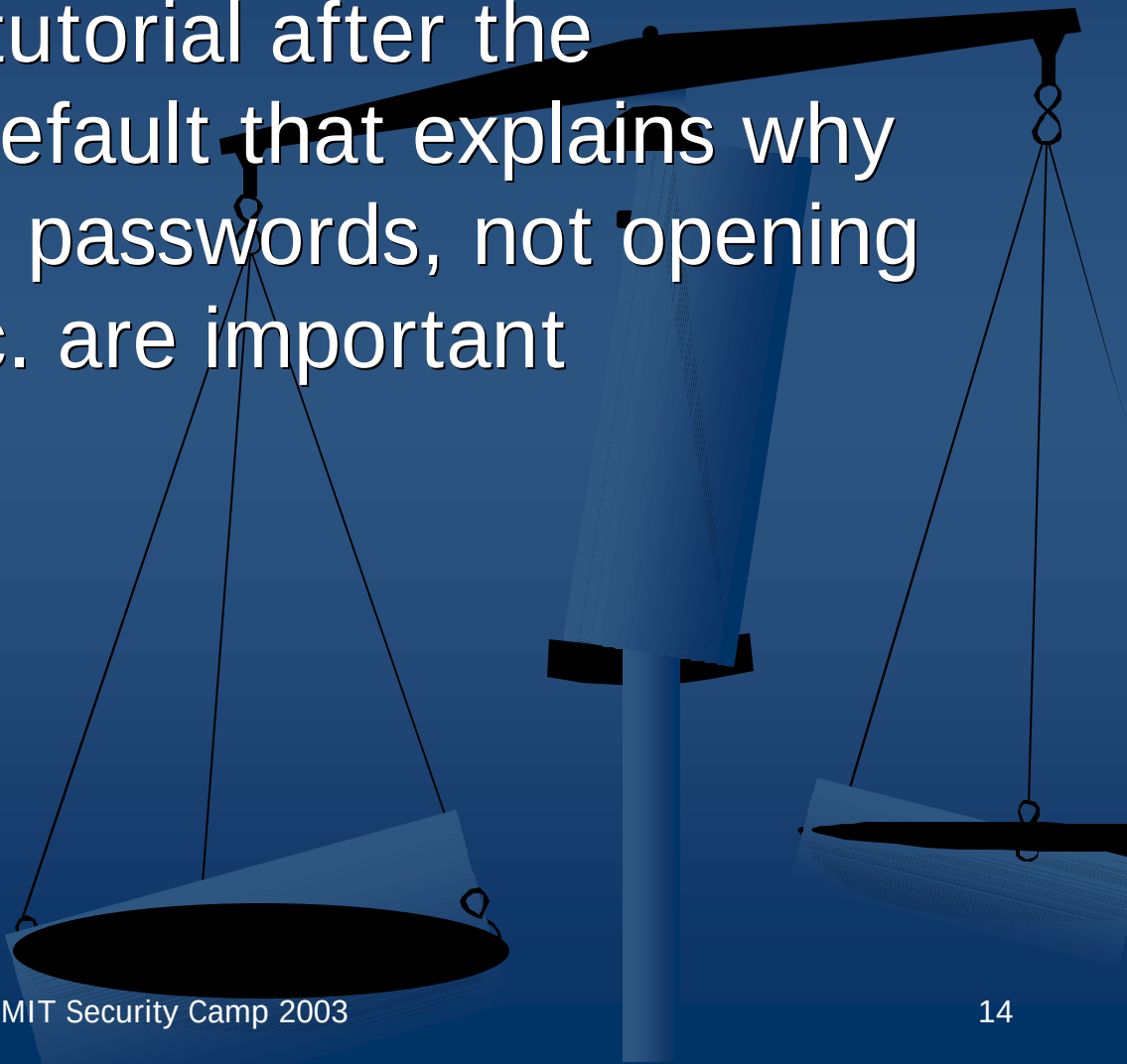
Warn if AV software not installed

- Check if VirusScan is up to date and running
- Warn if it isn’t and direct to further help



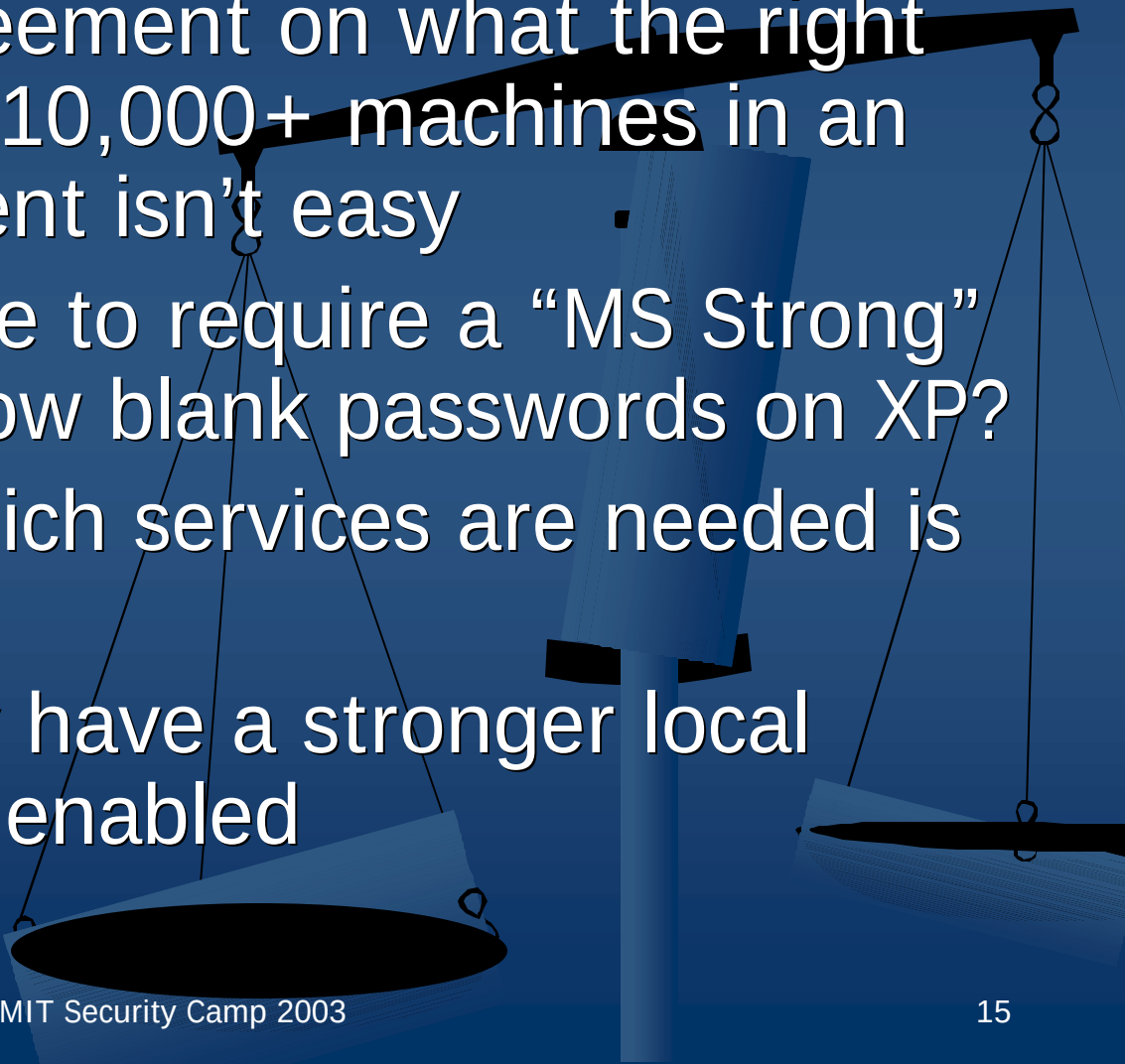
User Education

- Provide a brief tutorial after the installation by default that explains why things like good passwords, not opening bad e-mails, etc. are important



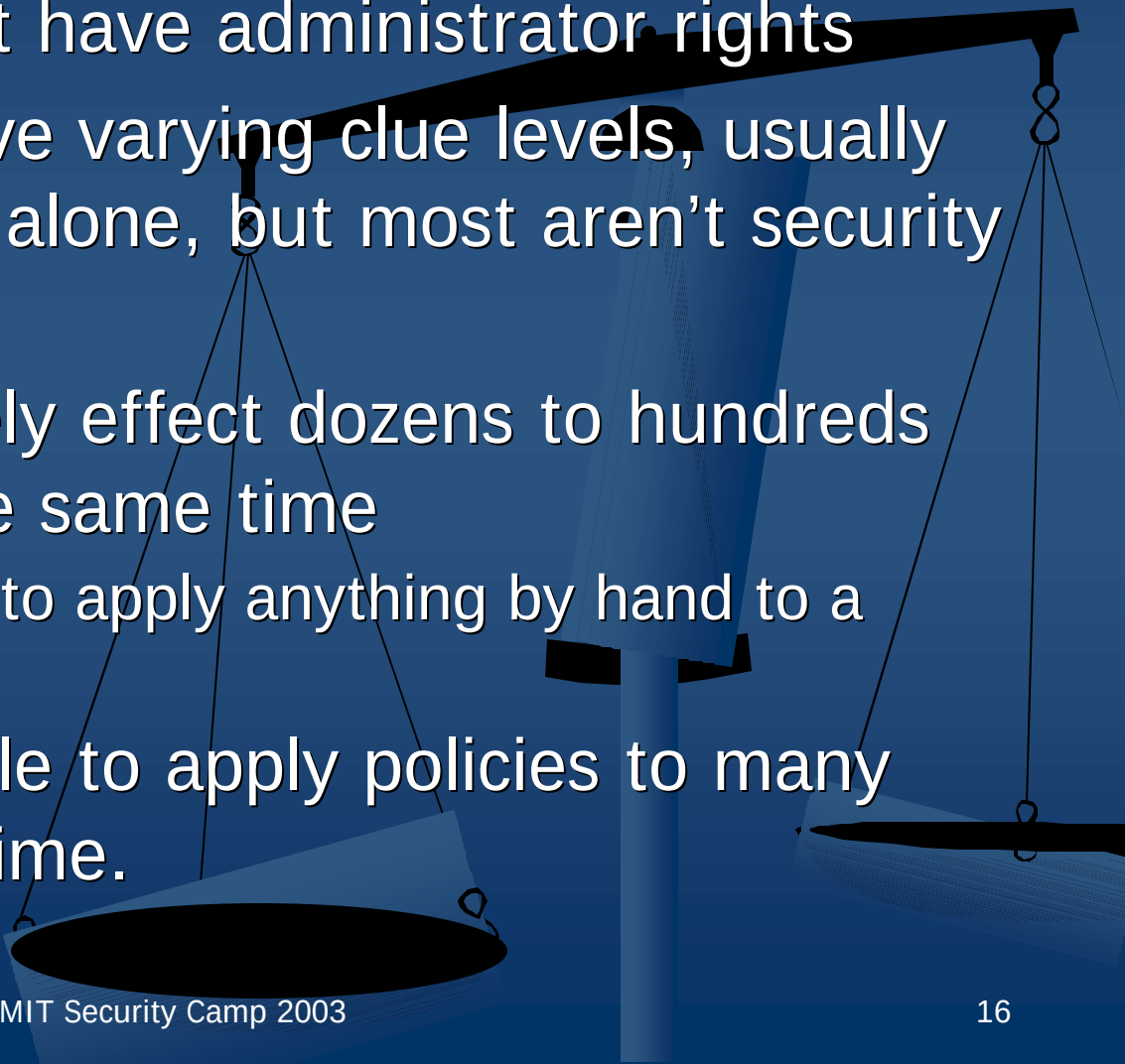
Challenges

- Developing agreement on what the right settings are for 10,000+ machines in an open environment isn't easy
- Is it more secure to require a "MS Strong" Password or allow blank passwords on XP?
- Determining which services are needed is difficult
- Some folks may have a stronger local policies already enabled



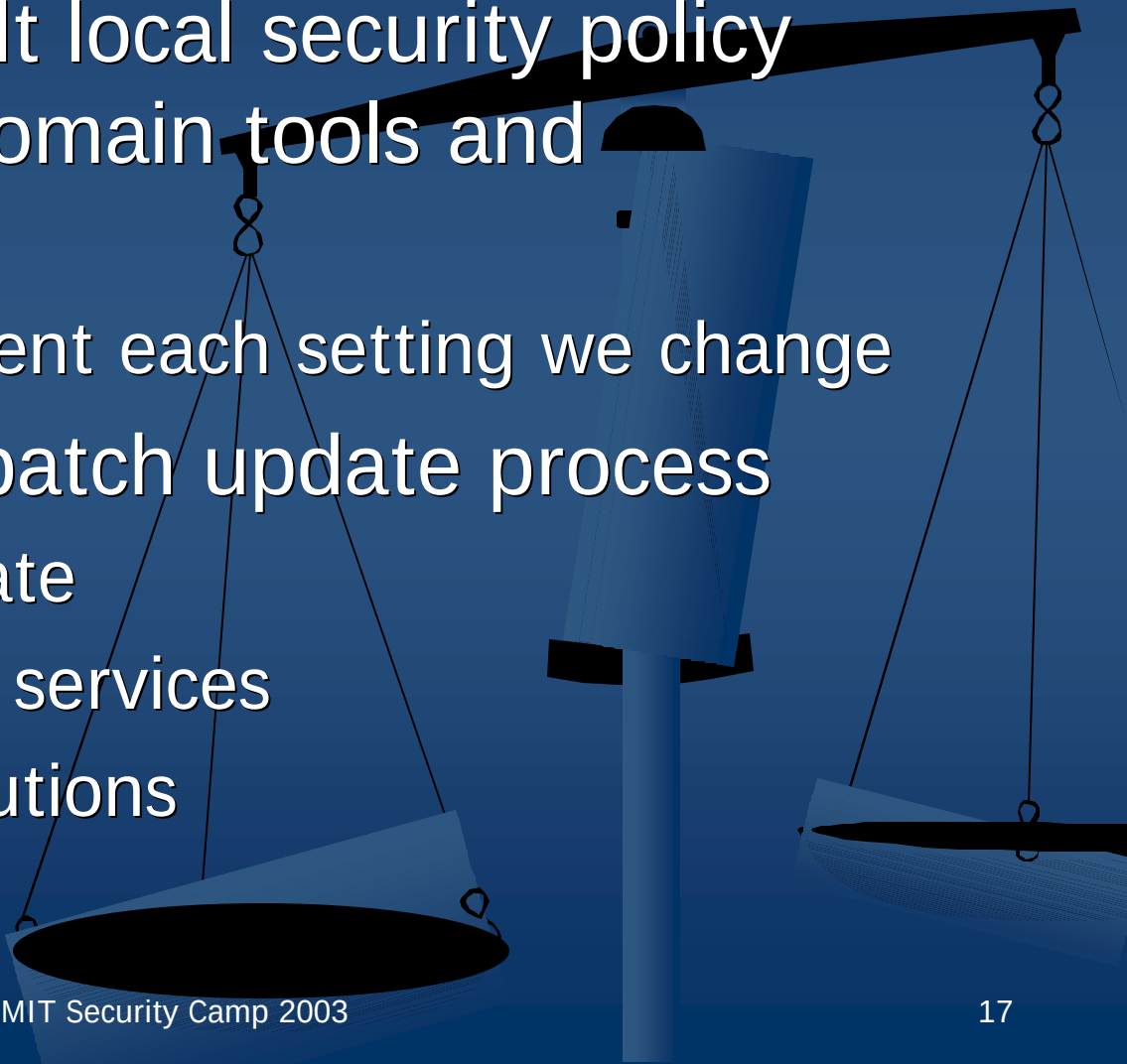
Difference with Domains

- Users often do not have administrator rights
- Administrators have varying clue levels, usually higher than stand alone, but most aren't security experts
- A problem will likely effect dozens to hundreds of machines at the same time
- It takes a lot of time to apply anything by hand to a hundred machines.
- More tools available to apply policies to many machines at one time.



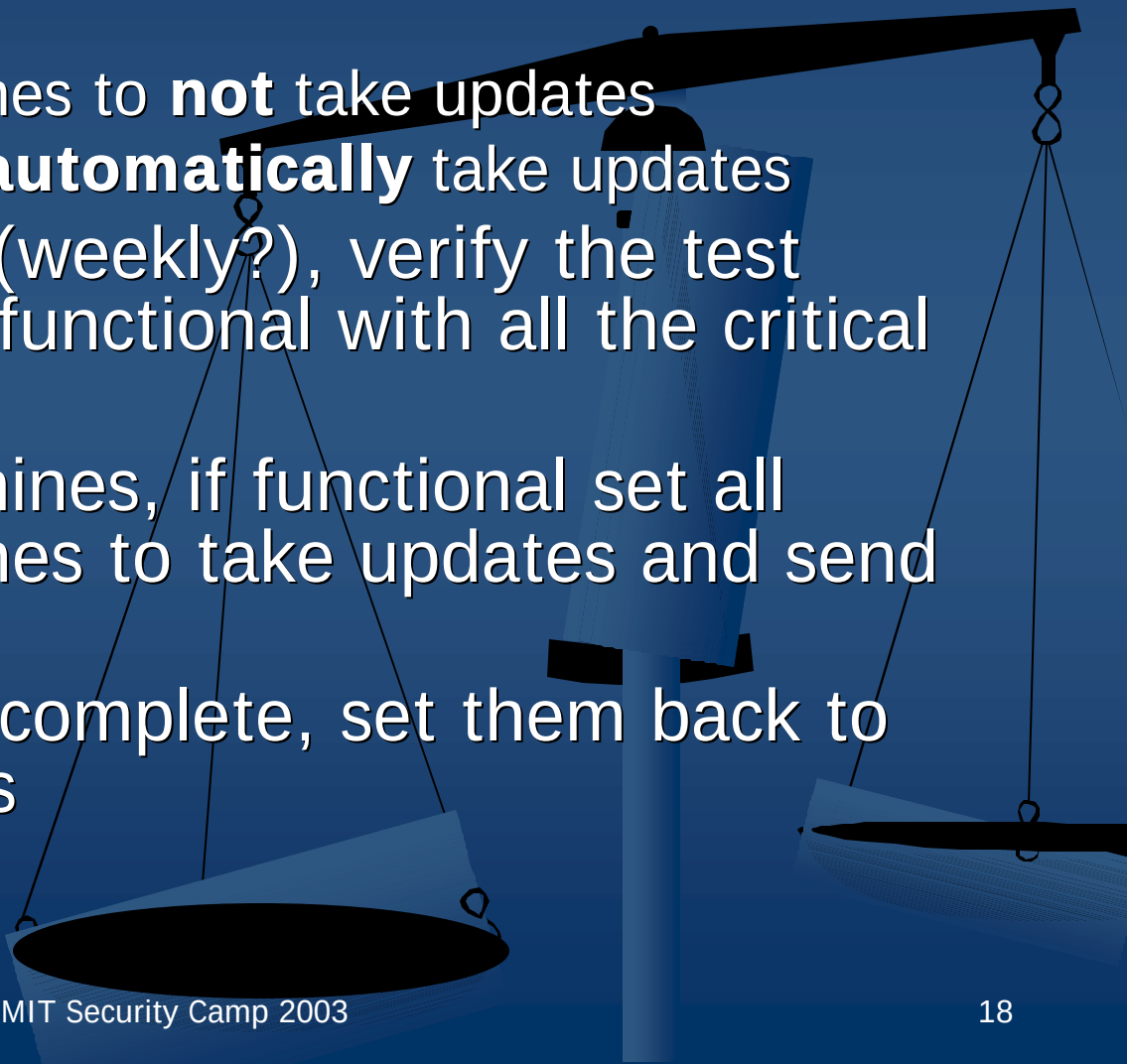
Managed NT 4.0 & 2000 Domains

- Provide a default local security policy applicable via domain tools and customizable
  - Need to document each setting we change
- Recommend a patch update process
  - Using AutoUpdate
  - Using Microsoft services
  - Using other solutions



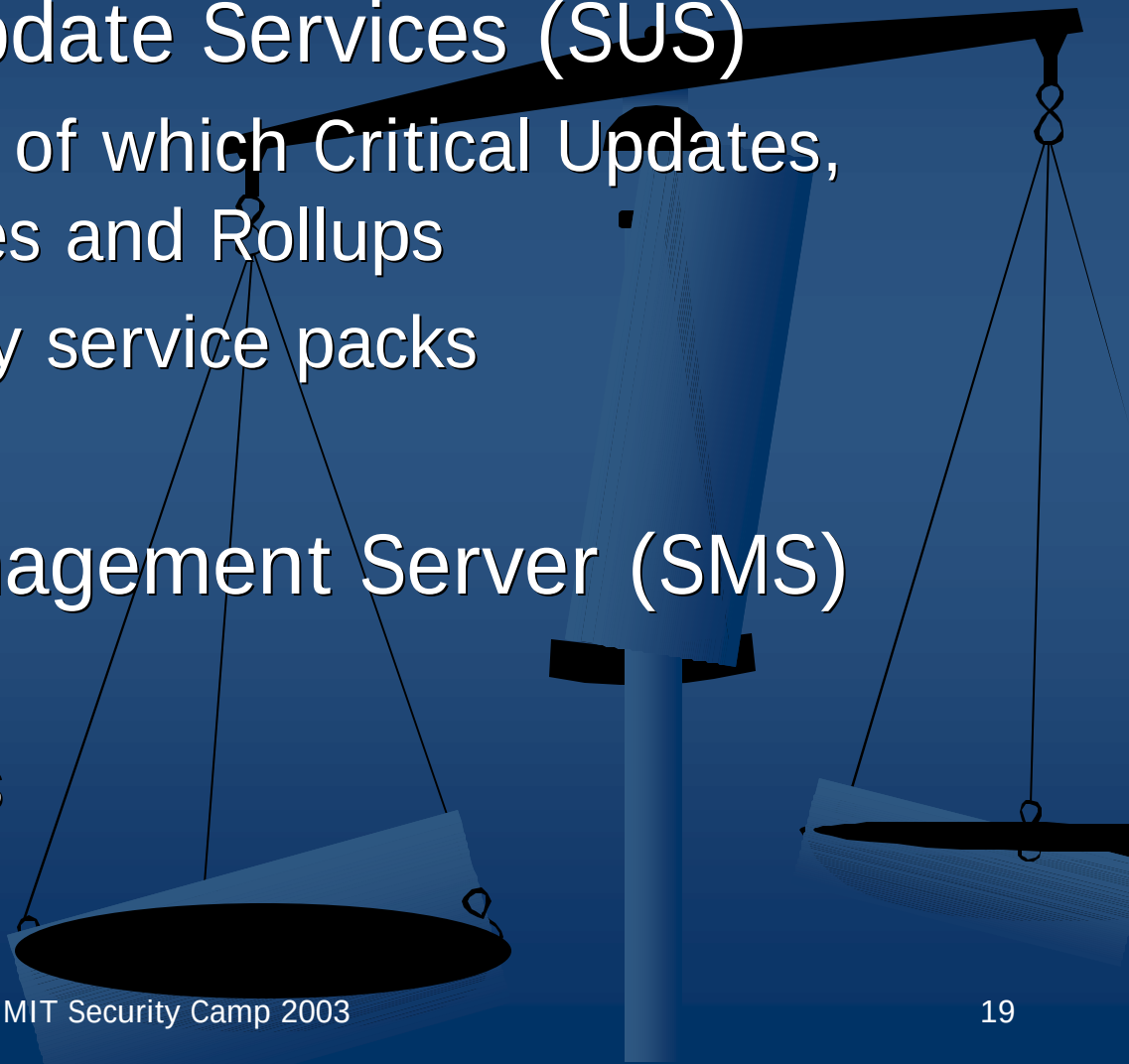
Domain Updates using AutoUpdate

- Configure
  - production machines to **not** take updates
  - test machines to **automatically** take updates
- On periodic basis (weekly?), verify the test machines are still functional with all the critical updates applied
- Validate test machines, if functional set all production machines to take updates and send reboot command
- After updates are complete, set them back to not taking updates



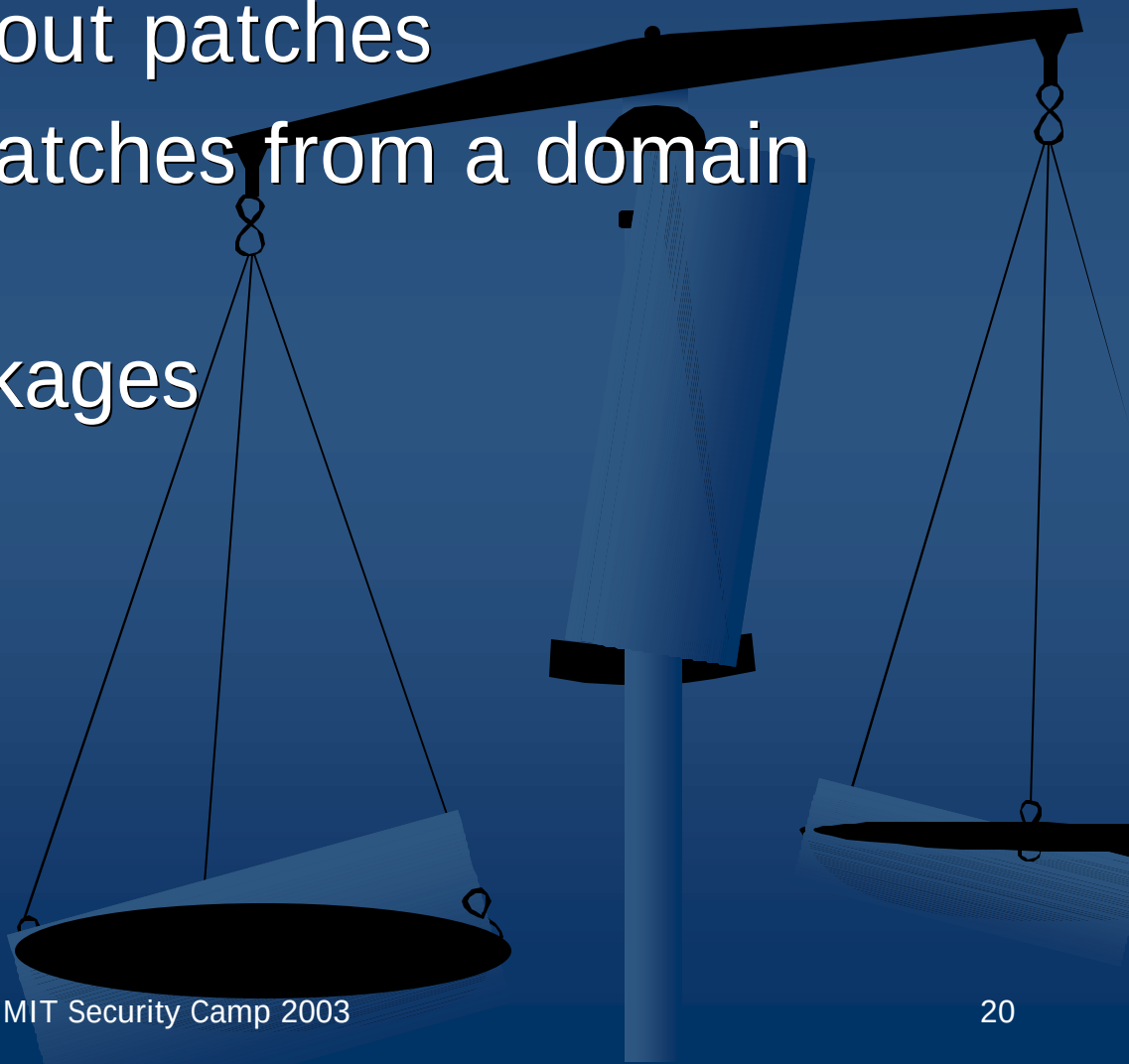
Domain Updates using Microsoft services

- MS Software Update Services (SUS)
  - Enables control of which Critical Updates, Security Updates and Rollups
  - Does not deploy service packs
  - Requires IIS
- MS System Management Server (SMS)
  - Granularity
  - Reporting Tools



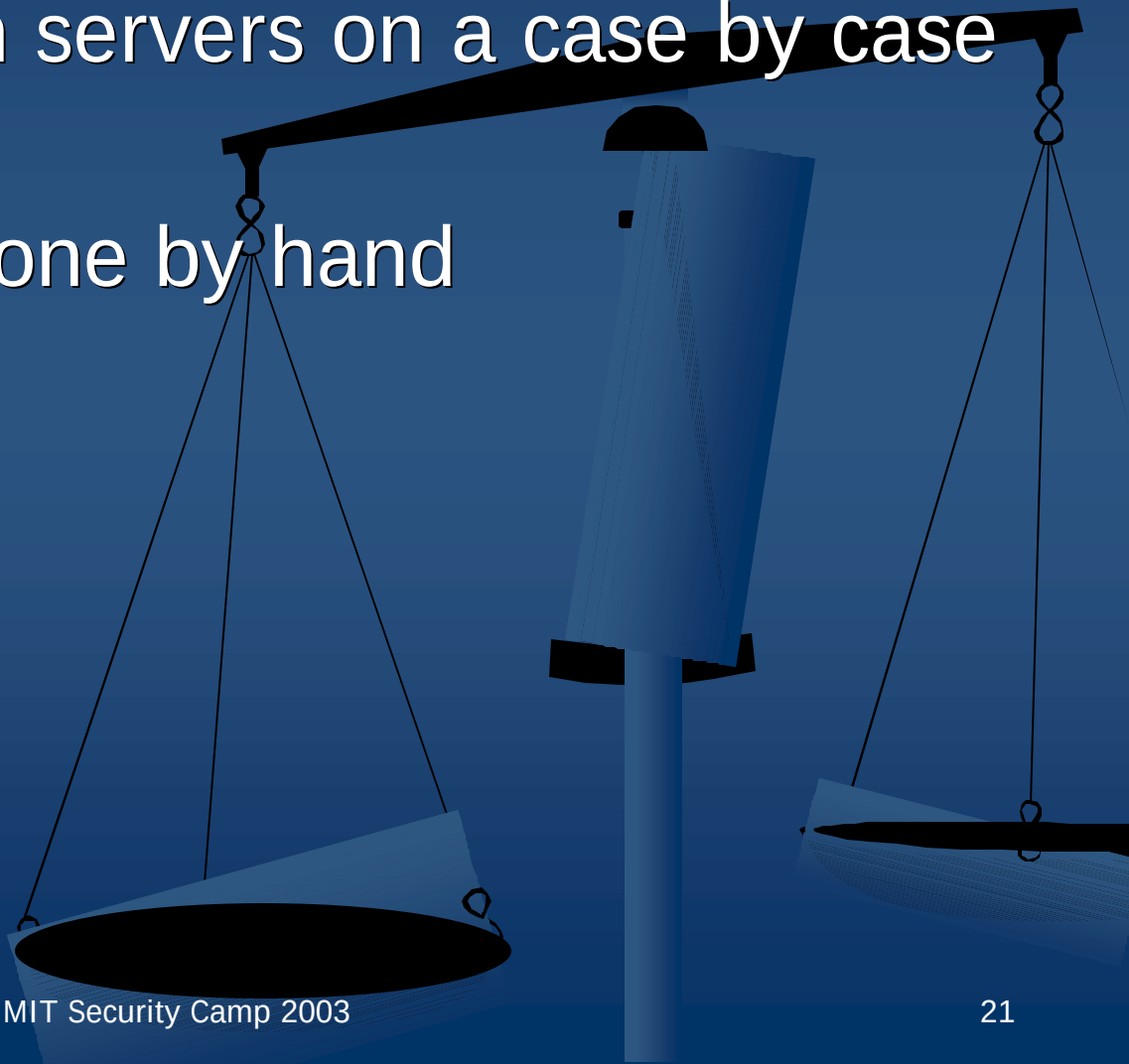
Domain updates using other solutions

- Scripts to push out patches
- Scripts to pull patches from a domain central location
- Third Party packages



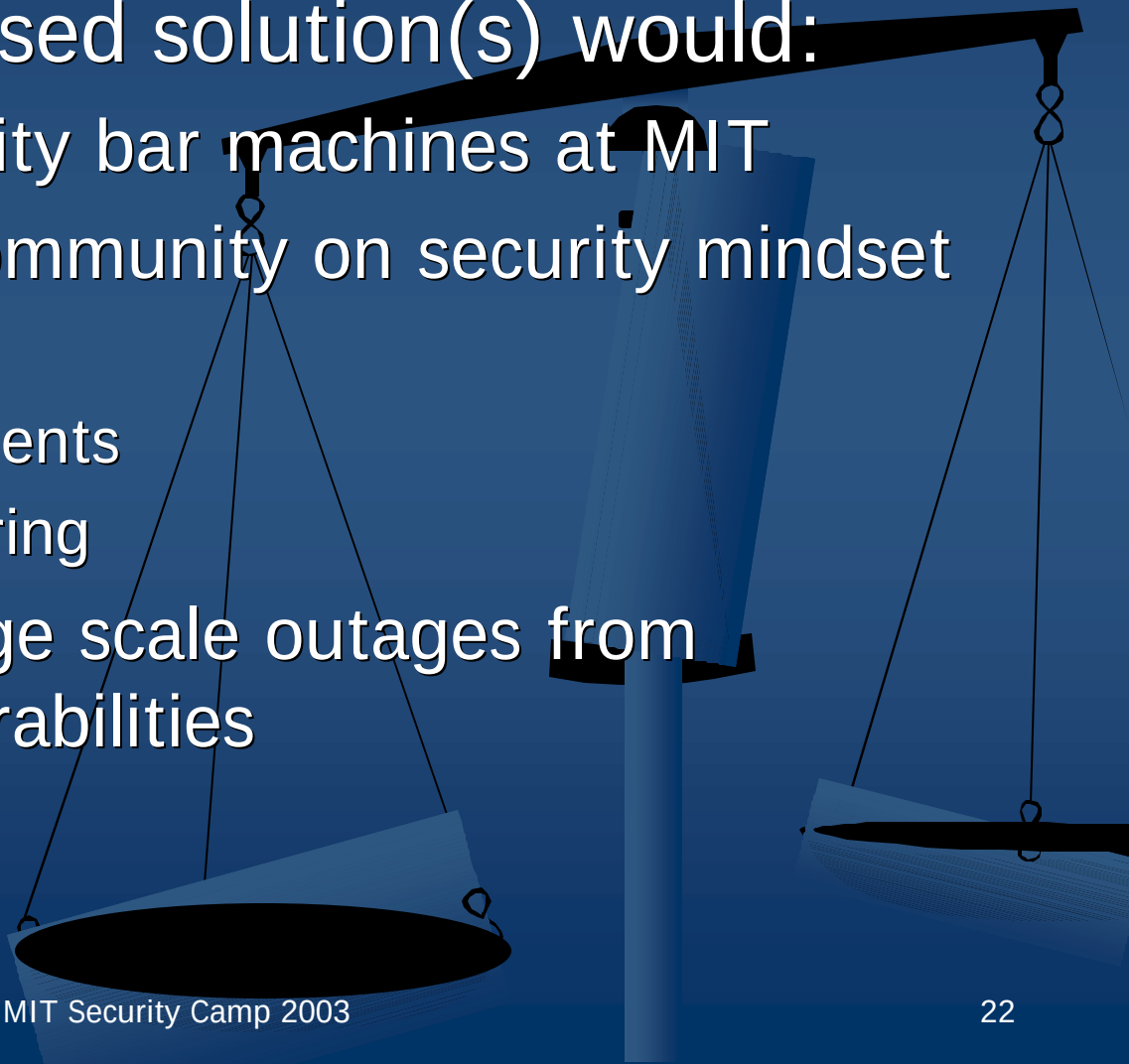
Domain servers

- Got to deal with servers on a case by case basis
- Probably best done by hand



Conclusion

- Hope the proposed solution(s) would:
  - Raise the security bar machines at MIT
  - Educate MIT community on security mindset
    - Passwords
    - E-mail attachments
    - Social Engineering
  - Prevent the large scale outages from exploited vulnerabilities



Questions/Suggestions?

jmhunt@mit.edu

Useful Links on SUS & SMS

- http://www.microsoft.com/windows2000/windowsupdate/sus/default.asp
- http://www.microsoft.com/windows2000/windowsupdate/sus/susfaq.asp
- http://www.microsoft.com/smserver/evaluation/datasheets/PatchDeploy.asp