AD Policy Working Group
May 2nd, 2014
(shifted from April 18th, 2014)
B3 Hillsborough
3pm-4:30pm

Voting Members: Donna Barrett, ~~Charles Cline~~, Dan Green, Julie Tilley, ~~Dan Evans~~, Daniel Henninger, Tom Farwig, Joshua Gira, Payman Damghani

Ex Officio Members: ~~Alan Gerber~~, Michael Underwood, ~~Gene Morse~~, Jonn Perry

Guests: Billy Beaudoin

Business handled outside of meetings:

- **Updated SCCM Service Team Leads to the AD Policy mailing list (10/21/13)**
  As a part of the creation of the SCCM Service teams I'm going to be proposing that we add the leads of each group to the AD Policy mailing list (and google docs). While they won't have a vote, they will need to be able to participate in the email discussions and offer feedback just as they would at the physical meetings. While we're still getting the service teams sorted out, I've gone ahead and added Alan Gerber and Michael (two of the proposed leads) so we can continue discussing the SCCM / Powershell issues we had on the last agenda.

  Updated 11/08/13 -- added Gene Morse, proposed lead for SCCM Core Service Team.
  Updated 02/24/14 -- added Jonn Perry, proposed lead for the SCCM Patching Service Team.

- **Released MS FixIt script to head off Snowman exploits (2/28/14)**
  Detailed in https://sysnews.ncsu.edu/news/5310a679 -- with the recommendation and request of the NCSU Security subcomm, the AD Policy approved the immediate push of the MS FixIt to address this exploit in the wild.

  ACTION: Need to check and confirm if we've stopped pushing this out to computers. No need to remove it, just not reason to keep pushing.

- **Reviewed Final Draft of SCCM Service Team Proposal (3/3/14)**
  "...designed a model of self-managing service teams to handle various aspects of the service. The proposal includes a core team, application packaging team, imaging team, and patching team, as well as a service coordinator role to act as a point of contact for the service. AD Policy will be the governance body that will approve this plan and approve changes to it. AD Policy will also approve the person who fills the service coordinator role. AD Technical will approve the creation of new service teams…" Committee Members had until 3/14/14 to offer comments / suggest changes to the draft.

- **OnBase Added to Campus Trusted Sites collection (3/20/14)**
  We've received a request from Brian Fontaine of OIT-Shared Services to add the following to the "Trusted Sites" collection within the central "WolfTech-Default Domain Policy - Desktop OS" group policy.

  https://obprd.acs.ncsu.edu

  Required for campus clients to support the upcoming OnBase go live date. OnBase is the imaging system that will be replacing Singularity. So every time you request an image from the FIN system (and other systems going forward) through Peoplesoft, that image will come from https://obprd.acs.ncsu.edu.

  Committee Approved 3/21/14 and issue was passed to AD Tech Comm for review and implementation by the domain admins.

- **Request for rpt2web.acs.ncsu.edu in domain Intranet Sites (3/31/14)**
  Request by Wade Davis of OIT-EAS to add rpt2web.acs.ncsu.edu to the IE Intranet sites domain level policy (WolfTech Default Domain Policy - Desktop OS). The domain rpt2web.acs.ncsu.edu hosts the Report2Web application that provides web-based report distribution and approval for the administrative applications, including the HR System and Financials, to all of campus. The current version is only functional in IE when the Compatibility View mode is enabled, so we would like to have this site added to the Intranet Zone in IE so that Compatibility View will be used and users do not experience problems with Reports2Web.

  [Dan's note -- one of the side affects of putting a site in the Intranet Zone is that compatibility mode is used by default.]

  Committee Approved 4/1/14 and issue was passed to AD Tech Comm for review and implementation by the domain admins.

=====

Agenda:

SCCM Membership Query (Michael)
I've noticed a lot of slowness on collection updating including ones with direct memberships. Sometimes taking upwards of 30 minutes for new direct members to show in the collection. This has been brought up in the past to me by Ryan, who suggested that having the Update incremental updates for this collection checked might be the problem. We know in our environment it doesn't work for a number of reason, and I didn't think of it as being much of a problem since my thought processes was it currently doesn't work in our environment but it might in the future so I don't think it's a big deal if it's checked. While having update incremental

updates for this collection doesn't actually update the membership for our collection it is still querying the database every 15 minutes to try to update it's membership.

We currently have 924 collections with that setting checked. So every 15 minutes 924 collections are querying the database trying to update it's membership and failing on top of the full discoveries that are running almost continually. From various articles online Microsoft recommends only having about 200 collections total with that checked, and we are well over that recommended limit.

To fix this in Powershell is very simple.

```
$Col = Get-CMDeviceCollection | Where-Object {$_.RefreshType -eq 6}
ForEach ($Obj in $Col)
   {$Obj.RefreshType=2
   $Obj.Put()
   }
```

Something as basic as the above script will take care of things. I know Alan wrote a more complex script to do the same thing. I would recommend either using my script or Alan more robust script to uncheck the box for all collections.

In the interest of statistics, all of our collections within SCCM fall into one of these four categories:

- "Good" collections (no incremental updates flag; 2+ hour refresh intervals): 486
- Collections with Incremental Updates only enabled (no repeating update schedule specified): 0
- Collections with both Incremental Updates and 2+ hour refresh intervals: 909
- Collections with too-frequent refresh intervals specified (less than 2 hours) and no Incremental Updates enabled: 7

Update from Allen (4/29/14):

SCCM Cron Server: 90% done
The cron server is configured via group policy to pull scripts from a DFS path that is restricted to members of the SCCM Core Service Team. It has the requisite permissions in place to allow those service team members to configure cron scripts to execute without needing remote desktop permissions. All cron scripts will be set up to run as the SCCM Cron Service Account rather than our own individual accounts. The remaining work that needs to be done with this server is adjusting the ComTech firewall rules to allow WMI remoting between the cron server and the site server [more on this below]. Additionally, once this work is complete, the last of the testing of the cron service account can take place.

removePNDUscope.ps1: 80% done
This is the first script I started working on. Originally supposed to be only about 6 lines of code, it was a very obvious piece of simple, low-hanging fruit. It removes the "Placeholder-Do Not Use" scope from SCCM application and package definitions; this is necessary because our environment associates that scope with all newly-created software objects, and many authors forget to remove that scope from their objects, displaying the object in the console of all other users with authorship rights, and also leaving the object open for such users to edit and modify, a potential security risk. The idea is to run this script on a daily schedule to mitigate that security risk and declutter the admin console for everyone. Unfortunately, a closer reading of the various powershell cmdlets available revealed that manipulating the scope membership of package definitions is much more complicated than application definitions. While the application definition portion of the script is complete, the package portion is being rewritten to accommodate the quirks of the interface for the package model. The script was last run manually on April 21st, successfully manipulating the application model objects to remove the PDNU scope. It has since been running daily as a cron, only manipulating the application model objects, as my own .ccmadmin account. Once the package model coding rework and the cron server work has been completed, I intend to switch this to being executed as the cron service account.

setCollectionRefreshSchedules.ps1: 100% done
I'm pleased to report that this script was successfully executed this afternoon and removed the incremental updates flag and reset collection membership update schedules to 2 hours in the case where collections were set to update more frequently. I've observed that the collection evaluator in SCCM, while still busy, is significantly less so, and manual collection membership update requests that I've submitted since that execution have generally taken place within about 10 seconds of the request. The script must take these actions by WMI, and currently the ComTech firewall is preventing such communications between the site server and the cron server. Thus, for the time being, the script must be run manually from the site server itself. When I return, I'll coordinate with ComTech to get the appropriate firewall exceptions added and configure this as a daily cron.

DASA Account Request for NCSU-Read Group Memberships (Charles)

DASA Technology is using a web-based product called GLPI to track inventory information. We are currently using the local authentication method GLPI offers, however, this method is simple and doesn't give us the ability to implement best practices in identity and access management we need. We would like to use an AD service account (DASA.glpi.service) with permissions to read the "memberOf" attribute of AD objects. With this we can create an AD security group that contains the user accounts allowed to log in to our instance of GLPI. This will match many of our other methods of service access within DASA. We can also leverage existing AD policies to enforce password complexity and change intervals. GLPI's built in authentication method doesn't provide any options for password enforcements. (justification provided by Doug Flowers)

ACTION: **Committee Approves.** Will pass to AD Technical Comm for review and implementation by Domain Admins.

Windows 8.1 Baseline Issue w/ Remote Assistance (reported by Chris Bettini)
From a Win8.1 pc I could no longer send the "offer" to a Win7 or a Win8 computer after the baseline updates (see https://sysnews.ncsu.edu/news/53445925). From the Win8 pc it looked like it was working but the prompt would never show up on the computer I was trying to assist. The msra.exe service would run on the destination computer but they never got the prompt asking them to allow me to remote assist. Windows 7 was working correctly and from a Windows 7 pc I was able to use remote assistance to assist on Windows 8 just not the other way around.

Troubleshooting performed by Chris:
I blocked inheritance on my test ou and then linked all of my gpo's to the test ou. I then tested remote assistance again and everything worked correctly from Windows 7 and 8. I then made a copy of the "Win8.1 computer policy (SCM, v1.0)" and linked it to my test ou which gave me the initial problem again. I started removing certain settings from my copy of "Win8.1 computer policy (SCM, v1.0)" until I found the one that was causing the remote assistance issue. The setting "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" set to enabled was the culprit. I changed it to "not defined" and that fixed the issue. I set my test ou back to normal and then just added "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" to disabled on my CVM-Win8.1 policy and I'm working normally now.

Recommendation: Change "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" set to not defined. This setting is not defined in the Win7 or Win8 baseline policies.

ACTION: Committee Approves and requests Domain Admins update policy.

Self Service and Mandatory Deadline (Michael)
I'm not sure if anyone has run into this as of yet but when you make an application "Available" and that application Supersedes other Applications when you create the deployment on the Deployment Settings screen there is a checked grayed out option call "Automatically upgrade a superseded versions of this application", then on the next page "Scheduling" at the bottom you see "Installation deadline to upgrade users or devices that have the superseded application install:" and by default it is set to "As soon as possible after the available time".

I'm cool with leaving these as is, but the whole point of a Self Service applications is to allow end users to install when it is best for them, but if Superseded applications are going to upgrade as soon as possible then it's not very Self Service.

What should we do about the deadline of Applications and Superseded software? 1 year? 2

years? 6 months?I looked the the "best practices" guide and didn't see any mention of this. (as it turns out, this solution doesn't work -- as proven with Firefox issues earlier this year).

Dan's random thought: script to look at the versions that a computer is assigned to and report to OU Admins when a computer is in multiple versions? Help cleanup?

Billy: differentiate research apps from productivity applications.
Daniel H: CHASS is used to this sort of behavior and in some cases, relies on it…

ACTION: Need to document this behavior on the AD site -- but then also have a list of the few exceptions to this. Note how an exception is made. Should also then be communicated out to the list. But no change in behavior.

Windows XP
Discussion regarding the upcoming May 12th deadline.

IE Bug Out of Band Patch Released
There's been a lot of concern (https://sysnews.ncsu.edu/news/535ffebe) with a new IE bug that affects all versions of Internet Explorer. Microsoft has released an out-of-band security patch (http://blogs.technet.com/b/msrc/archive/2014/05/01/out-of-band-release-to-address-microsoft-security-advisory-2963983.aspx) that addresses this issue that the new SCCM Patching Service Team should be reviewing, announcing, and releasing per the domain patching policy (http://activedirectory.ncsu.edu/services/patching/). ***Released to campus 5/2/14.***

New AV Rollout (Payman)
AV servers will distribute the load via major AD OUs. (all of ENGR on one server, CHASS on another, etc). There are two components -- network client and the actual scanning agent. All machine will talk initially to a single server -- clients will then be passed off to their correct server.

But there's still some issues with the AD polling that's being resolved.

Components they want to install -- (firewall and encryption portions will not be installed) heuristics, file, websites, IM attachments, network attacks, email scanning (IMAP clients, not gmail). What will be on by default? File, malware heuristics, email scanning, and web will be on by default but could be turned off -- this would be as low as local admins. Of course, dept IT could disable this…

Timeline? Waiting for vendor to resolve current polling issue.
(we have until July 1st deadline to complete the migration)

VMs hosted on the OIT vCenter will be excluded from using this client as they have an alternative.