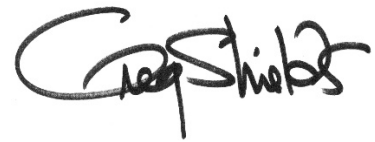


TechMentor – Absolute Beginner's Guide to Advanced ADCS

***** -- Notes to the Reader

- What you're currently holding are my notes for the TechMentor session noted above. As my notes, they're my own personal reminders about all the steps necessary to make our way through the demos. As I've been asked many times for these notes after this session, I provide them to you here as they are, warts and all. While I offer no support for them once the conference concludes, I wish you great success in building your own AD CS infrastructure!



***** -- Definitions

- **Enrollment agent.** An enrollment agent is a user who can enroll for a certificate on behalf on another client. Unlike a certificate manager, an enrollment agent can only process the enrollment request and cannot approve pending requests or revoke issued certificates. When you create an enrollment agent, you can further refine the agent's ability to enroll for certificates on behalf of others by group and by certificate template. For example, you might want to implement a restriction that the enrollment agent can only enroll for smart card logon certificates for users in a certain office or organizational unit that is the basis for a security group.
- **AIA's.** Authority information access locations are URLs that are added to a certificate in its authority information access extension. These URLs can be used by an application or service to retrieve the issuing CA certificate. These CA certificates are then used to validate the certificate signature and to build a path to a trusted certificate.
- **CRLs.** CRLs are complete, digitally signed lists of unexpired certificates that have been revoked. This CRL is retrieved by clients who can then cache the CRL (based on the configured lifetime of the CRL) and use it to verify certificates presented for use.

The legal stuff...

THE CONTENT OF THESE NOTES IS DISTRIBUTED IN THE HOPE THAT IT WILL BE USEFUL, BUT WITHOUT ANY WARRANTY. IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THIS CONTENT IS WITH YOU. SHOULD THEY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW THE AUTHOR WILL BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THIS CONTENT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF THE AUTHOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

- **CDPs.** CRL distribution points are locations, typically URLs that are added to a certificate in its CRL distribution point extension. CRL distribution points can be used by an application or service to retrieve a CRL. CRL distribution points are contacted when an application or service must determine whether a certificate has been revoked before its validity period has expired.

***** -- Introduction

- Sort of surprising (and yet not surprising) how few ADCS infrastructures you **don't** see in IT shops these days.
- Good news: Building your own in-house ADCS PKI isn't as difficult as you'd think.
- Hardest part: Figuring out how to decypher the install. Just too many articles with too much information.
- We'll talk about a minimum install that's easily doable. Get you off the ground.
- Worth mentioning that a three-tier structure for MOST PEOPLE is overkill.
- Microsoft recommends against a single-server infrastructure. If you have any problem, you'll have to recreate the entire infrastructure if you want to generate more certificates.
- A two-tier structure is just fine for most organizations, and so that's the configuration we'll walk through today.
- You can simply repeat the process we'll see here to create that three-tier structure.

***** -- Configure offline root CA

- ROOTCA should be in WORKGROUP, but we're installing in domain here (against best practices) just to simplify some of the demos.
- ROOTCA and ISSUINGCA have Cert Services installed.
 - ROOTCA only has Certification Authority role service.
 - ISSUINGCA has all role services.
- ROOTCA | Configure Certificate Services.
 - Standalone CA | Root CA
 - Create new private key
 - (IMPORTANT) 4096 bit key length
 - WARNING, SERVER CORE GOTCHA: This checkbox enables private key strong protection. This means that you will have to enter administrator password or confirm action each time private key is used. It will be used each time when new certificate/crl is issued. And when service starts. If you are using HSM, you should consult with HSM documentation to determine whether your HSM requires this setting. As you don't use HSM by now, then you should not enable this checkbox, because you won't see any prompt dialog on server core.
 - SHA256 (NOTE that SHA-1 certs will be blocked by Windows on 1/1/2017, SHA-1 was deprecated back in late 2013)
 - Validity period : 5 years (choose your poison)
 - As a general rule, the validity period of CA certificates should be at least twice as long as the maximum validity of the certificates issued by the CA. For example, if a CA issues certificates that are valid for two years, the validity period of the CA certificate should be at least four years. A CA certificate is usually renewed in

the middle of its lifetime so that it can keep issuing certificates during the full validity period. The recommendation is to renew the CA certificate once while keeping the same key pair and to renew it again while changing the key pair.

- Show all tabs, then talk about CDPs and CRLs.
- ROOTCA | Properties of CA | Extensions
 - Machines using these certificates need to be able to see the CRL. So, we need to publish it to an available location on the ISSUINGCA.
 - Reconfigure to publish CRL to
<http://issuingca.company.pri/tacoburrito/<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl>
 - Include in CRLs, and in CDP extensions. Not in IDP extensions.
 - IDP extensions are used by non-Windows clients to verify revocation. Allows for “partitioned CRLs” where 3rd party CAs publish different CRLs for different certificate types (user/computer/etc).
 - Note that we haven’t created this folder yet. We’ll do that shortly.
 - Remove CRL LDAP references.
 - Repeat the process with the AIA:
<http://issuingca.company.pri/tacoburrito/<ServerDNSName> <CaName><CertificateName>.crt>
 - Include in AIA extension (not OCSP extension).
 - Remove AIA LDAP references.
 - Force a publication of the CRL.
 - Navigate to [\\ROOTCA\\c\\$\\windows\\system32\\certsrv\\certenroll](\\ROOTCA\\c$\\windows\\system32\\certsrv\\certenroll). Show the CDP/AIA files there.
- Create a new certificate out of the root certificate.
 - Properties of CA | General Tab | View Certificate | Details | Copy to file | .CER file |
[\\ROOTCA\\c\\$\\windows\\system32\\certsrv\\certenroll\\rootcacert.cer](\\ROOTCA\\c$\\windows\\system32\\certsrv\\certenroll\\rootcacert.cer).
- Gpupdate /force (to acquire root CA cert, this environment only).

***** -- Deploy Root CA certificate using Group Policy

- Offline CA servers won’t necessarily have their certificates automatically delivered through Group Policy. Our environment has the Root CA online (bad practice, but it’s a lab). So, let’s show where you can configure one, just in case.
- Show MYDESKTOP | Trusted Root Authentication Authorities. Note that there is no CA cert in the list.
- Show how the cert can be manually entered by double-clicking the .CER file. Don’t actually install the cert.
- Create a new GPO | Computer | Windows Settings | Security Settings | Public Key Policies | Trusted Root Certification Authorities | Import certificate
- Gpupdate /force
- Refresh certificates view and show ROOTCA certificate on MYDESKTOP.

***** -- Configure an Issuing CA

- ISSUINGCA | gpupdate /force
 - To get the root certificate installed locally.
- ISSUINGCA | Configure Certificate Authority role service | Enterprise | Subordinate | New Private Key | Request a certificate from parent CA.
- Navigate to C:\ and show certificate request.
- ROOTCA Certification Authority console | All tasks | Submit new request | Request file.
- Show pending requests | All tasks | Issue.
- Issued certificates | Export as P7B file. Show that the CDP has been configured to point to ISSUINGCA on the root cert.
- Transfer P7B file to ISSUINGCA.
 - NOTE: At this point we can't yet install the certificate, because ISSUINGCA can't resolve the CDP/AIA location we populated in that cert. We need to first create that folder.
- ISSUINGCA | Create folder c:\inetpub\wwwroot\tacoburrito. Then, copy the content from ROOTCA\c\$\Windows\System32\Certsrv\CertEnroll to this location.
- IMPORTANT: Allow for double-escaping (due to plus sign in Delta CRL).
 - IIS Manager | Tacoburrito folder | Request Filtering | Edit Feature Settings | "Allow Double Escaping"
- Test functionality by opening IE and navigating to <http://issuingca.company.pri/tacoburrito/company-rootca-ca.crl>. The file should be downloadable.
- ISSUINGCA | Certification Authority console | All tasks | Install CA Certificate
- Start CA Service on ISSUINGCA.
- ISSUINGCA | General tab | view certificate | show certification path and that the ISSUINGCA cert was issued by ROOTCA.
- Power down ROOTCA.
- Run pkiview.msc to test config.
- ISSUINGCA | Certificates console | Show that ISSUSINGCA how has an ISSUINGCA cert in its Intermediate Certification Authorities | Certificates container.
- ISSUINGCA | Properties of CA | Extensions
 - NOTE: DO NOT publish CRL to a share on same machine. Only publish CRLs to non-share folder paths (e.g. C:\bleh).
 - Reconfigure to publish CRL to C:\inetpub\wwwroot\tacoburrito.
 - Then, configure CA to look for CRLs at <http://issuingca.company.pri/tacoburrito/<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl>
 - Include in CRLs, and in CDP extensions.
 - Remove other HTTP and FILE CRL references to CertEnroll (so clients look to one place only).
 - CRT files with AIA information CAN NO LONGER be published in non-default locations. So, the CRT file can only be published to C:\System32\Certsrv\CertEnroll.
 - So, ****COPY**** the CRT file to C:\inetpub\wwwroot\tacoburrito.

- (In the real world, this process needs to happen only when a CA cert is refreshed/changed/updated for any reason, which isn't all that often. Can create a separate script to do this, or just remember to re-copy the file.)
- Reprint the AIA to the HTTP location:
<http://issuingca.company.pri/tacoburrito/<ServerDNSName> <CaName><CertificateName>.crt>
 - Include in AIA extension. Not the OCSP.
- Remove other HTTP and FILE AIA references to CertEnroll (so clients look to one place only).
- Force a publication of the CRL.
- Navigate to [\\ISSUINGCA\\c\\$\\windows\\system32\\certsrv\\certenroll](\\ISSUINGCA\\c$\\windows\\system32\\certsrv\\certenroll) and C:\\inetpub\\wwwroot\\tacoburrito. Show the CDP/AIA files there.

***** -- Monitor CA health

- Launch pkiview.msc
- NOTE: If pkiview shows an error for OCSP, revoke the CA Exchange cert from ISSUINGCA and then run "certutil -cainfo xchg" to recreate the cert.

***** -- Configure Online Certificates Status Protocol responders

- A mechanism to allow computers to access the contents of the CRL without having to parse the CRL itself. Used in situations where the CRL is long to accelerate CRL checking. Greatly increases the performance in checking CRLs.
- Online Responder downloads the CRL itself, and is an optimized piece of software that can respond faster than fully downloading the CRL yourself.
- Note that you would need to regenerate any certificates that have already been generated and provisioned once you configure OCSP.
- Certificate Templates | Certificates | Rename OCSP Response Signing Cert
 - Change security on cert so ISSUINGCA\$ has Read/Enroll privileges
 - New Certificate Template to Issue
 - Online Responder Management
 - Create Revocation Configuration
 - Add Delta CRL to Revocation Provider configuration.
- Finally, to let the clients know to look for online responder.
 - Show the OCSP virtual directory in IIS. Talk about how this was created when we installed the online responder.
 - IF THIS DOESN'T SHOW UP, NEED TO RUN THIS COMMAND TO RE-REGISTER:
certutil -vocsproot
 - Properties of CA | Extensions tab | AIA | Create new http://<serverdnsname>/ocsp | Include in OCSP extension. (NOT IN AIA).
- Re-publish CRL and then re-copy .CRT file from C:\\Windows\\System32\\Certsrv\\Certenroll to C:\\inetpub\\wwwroot\\tacoburrito.
- Refresh pkiview.msc. Validate that OCSP is now working for all new certificates.

***** -- Enrolling for Certificates

- Let's now try enrolling for some certificates, using a couple of different methods.
- ISSUINGCA | IIS Manager | Enroll for web cert.
- ISSUINGCA | MMC, Certs | Enroll for computer cert.
- What about other certs, like code signing certs for PowerShell? For these, we'll need to customize a certificate template.

***** -- Manage Certificate Templates

- BE AWARE: You may not want to manage your own Authenticode "Code Signing" certificates, because AD CS does not include timestamp support.
 - Timestamping ensures that code will not expire when certificate expires. If your code is timestamped the digital signature is valid even though the certificate has expired. A new certificate is only necessary if you want to sign additional code. If you did not use the timestamping option during the signing, you must re-sign your code and re-send it out to your customers.
- Create a Certificate Template for Issue for PowerShell code signing. This requires a "Code Signing" template. But what actually MAKES a code signing template?
 - Cert Administrator | Manage | Copy of Code Signing | PowerShell Code Signing
 - Change validity period to 3 years.
 - Mention "Publish Certificate in Active Directory" - Used to prevent users from enrolling in multiple certificates. Prevents re-enrollment, often used for shared services like S/MIME.
 - Request handling tab - Purpose = Signature, "Allow Private Key to be Exported" (need if I'm moving my cert around to new machines)
 - Request Handling | Archive subject's encryption private key (greyed out here, but usable with other certificate templates)
 - Extensions tab - Issuance Policy
 - Extensions tab - Key Usage (each form has a purpose, and although some of the fields on forms are the same the purpose of the form may be different. Built into the certificate will be defined what the cert can be used for).
 - Extensions tab - Application Policies (This is Enhanced Key Usage)
 - Subject Name tab -
 - Issuance Requirements tab - (Mention renewal stuff at the bottom)
 - Security tab - Grant Read, Enroll, Autoenroll to IT Group
- New | Certificate for Issue
- Enroll for certificate on ISSUINGCA (this is a User certificate)
- Create a little PowerShell .PS1 file.
- Discuss use of Set-AuthenticodeSignature cmdlet
 - `$cert = Get-ChildItem cert:\CurrentUser\My -CodeSigningCert`
 - `Set-AuthenticodeSignature -Certificate $cert -FilePath C:\myScript.ps1`

***** -- Certificate Validation, Revocation, Security, and Policies

- Show "Issued Certificates" and Revoke Certificate | Set Reason Code to "Certificate Hold" (revocation is always permanent, except in the case of "Certificate Hold").
- Publish CRL
- Properties of Revoked Certificates | CRL Publishing Parameters (minimum of 30 minutes for Delta CRL publishing)

***** -- Manage Certificate Enrollment and Renewal to Computers and Users using Group Policy

- Configure all IT users to auto-enroll their code signing certificate
- Configure Public Key Policies
 - Certificate Services Enrollment Policy - Enable Auto-Enrollment
 - Auto-Enrollment Policy - Sets up the structures to enable certificates to be enrolled.
 - Cert template - Configure "Auto-Enroll" on cert for person.
 - Discuss renewal.
 - Show that GP is also where you can pre-populate certs using certs.
 - Some examples of certificates to enroll: Client Authentication, EFS, S-MIME, Smart Card

***** -- Configure and manage key archival and recovery

- NOTE: There is an "EFS Recovery Agent" and a "Key Recovery Agent". The former is for the direct recovery of data; the latter is for the recovery of keys that you give back to the user. You may use either/or/both of these when you enable EFS.
 - NOTE ALSO: Any user that attempts to encrypt a file in an AD where there is no EFS certificate in place will automatically create a self-signed certificate, which is exceptionally difficult to recover. So, either disable EFS via Group Policy or create the EFS and recovery agent certificates to protect them.
- Delete existing "Basic EFS" template
- Duplicate "Basic EFS template" - "Basic EFS Template with Archival"
 - Request Handling | Archive Subject's encryption private key
 - Configure Authenticated Users: Read/Enroll
 - New Template to Issue
- The process to recover a key involves possessing a Key Recovery Agent certificate (on the computer of the person doing recovery)
- Duplicate "Key Recovery Agent"
 - Security tab | Grant "IT" group Read/Enroll on cert
 - Issuance Requirements | Remove "CA certificate manager approval"
 - New Template to Issue
- Configure ISSUINGCA for key archival
 - Enroll for Key Recovery Agent certificate (this is a user cert)
 - Properties of ISSUINGCA | Recovery Agents tab | Archive the key
 - Select the user KRA certificate just enrolled
- MYDESKTOP | MMC | Users | Enroll for "Key Recovery Agent" cert

- Modify Properties of CA | Recovery Agents | Archive the key | specify the “Key Recovery Agent” cert just enrolled
- MYDESKTOP | MMC | Users | Enroll for “Basic EFS with Archival” cert
- CA Console | Add a column | “Archived Key” (shows which keys are archived)
- Refresh the column in the CA Console and show the Archived Key column. The Basic EFS with Archival cert should be in the list with “Archived Key” set to “Yes”.
- Two commands to recover keys (DO THESE ON ISSUINGCA)
 - Certutil -getkey gshields@company.pri recover.p7b
 - This process takes about 60 seconds.
 - The outputblob file is a PKCS #7 file containing the key recovery agent certificates and the user certificate and chain. The inner content is an encrypted PKCS #7 file containing the private key (encrypted to the key recovery agent certificates).
 - Certutil -recoverkey recover.p7b recover.pfx
 - This will recover the key out of the PKCS #7 file created, into a new .PFX certificate which you can hand to the user.
- For fun: Show EFS Group Policy options, and how easy this is now to configure.

***** -- Configure and manage Certificate Enrollment Web Services and Certificate Enrollment Policy Web Services

- Configure AD CS | Web Service & Policy Web Service
- Authentication type for CES: Username/Password
 - POSSIBLE USES FOR THIS:
 - Enrollment/renewal of certs outside LAN boundaries.
 - Enrollment/renewal of certs for non-domain joined machines.
 - Enrollment/renewal of certs for domain-joined machines that aren’t often in the environment.
 - Client certificate auth is new in Server 2012. This uses x509 certificates.
 - In our case here, use the Username/Password type, as an example of non-domain certificate enrollment.
- Further configuration required to enable outside clients to enroll/renew from this web server
 - Launch IIS on ISSUINGCA and show the CES/CEP folders now created.
 - Domain certificate request | Rebind HTTP to HTTPS and cert (Default web site | Bindings)
 - Single-click CEP website | Application settings | Friendly name = "Company.pri CA"
 - Adjust the “Enrollment ID”. Just change one of the characters in the GUID. I’ve already created an enrollment ID in this domain for another use, and they can’t be the same.
- Run IISRESET on ISSUINGCA.
 - Be aware that the CEP can take 30 minutes to poll for certificate templates. Use “iisreset” to accelerate this polling.
- Enroll a cert from a non-domain joined machine EXTERNAL
 - Add root CA to trusted root authorities on machine (prior to filming)

- Certificates console | All tasks | advanced operations | manage enrollment policies | Add
- Configure Policy server URI --> This will correspond to the CEP's .svc file on the IIS server
- `https://issuingca.company.pri/ADPolicyProvider_CEP_UsernamePassword/service.svc/CEP`
 - Authentication = username/password
- CA Console | Request new certificate | Company.pri CA enrollment policy
- Show Group Policy configuration: GPMC | Computer | Windows | Security | Public Key Policies | "Certificate Enrollment Policy Server"