Implementing Forefront Identity Manager 2010

Student Manual

Module 6: Managing Credentials with FIM

Table of Contents

Module 6: Managing Credentials with FIM 1
Module Overview 1
Lesson 1: FIM Password Management 2
Logon Names Versus Passwords 3
FIM Password Management Features 5
FIM Password Flow 7
Lesson 2: Password Self-service Reset 9
Configuration 10
Registration 12
Reset 13
Lockout and Unlocking 14
Lab 6A: Password Self-service 15
Lesson 3: Synchronizing Passwords – PCNS 16
PCNS Configuration and Flow 17
PCNS Considerations 18
Lab 6B: Configuring PCNS 19
Lesson 4: FIM Certificate Management 20
The Certificate Management Component 21
Steps to Implement the MA 22

Module 6: Managing Credentials with FIM

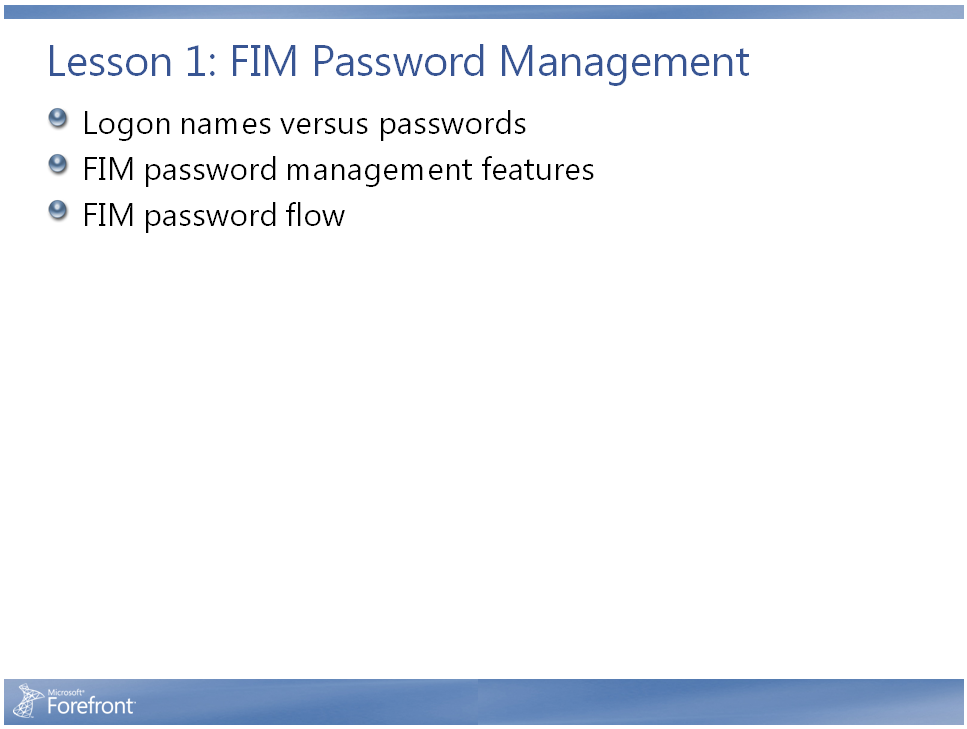
Module Overview

In this module we consider how Microsoft® Forefront™ Identity Manager 2010 (FIM) manages credentials. We are all familiar with using the basic credentials of logon/user name and password, but proper security increasingly demands stronger authentication features. When we speak of FIM, we are usually talking about the FIM Service and FIM Synchronization Service part of it, but the FIM brand includes FIM Certificate Management (CM), formerly called Certificate Lifecycle Manager (CLM). We do not cover FIM Certificate Management in this course because it is a massive topic on its own.

This module starts by considering user name and password management in general, and then focuses on Password Self-service Reset and Synchronizing Passwords.

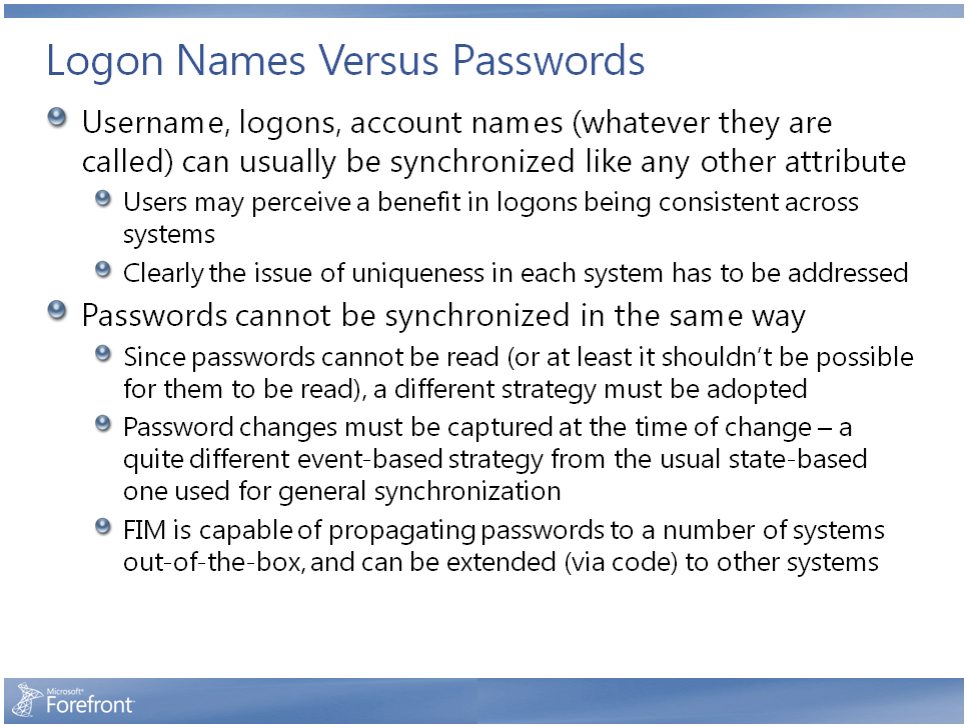
It also briefly touches on FIM Certificate Management (there is no lab for this last topic).

Lesson 1: FIM Password Management



This lesson looks at how, in principle, FIM can help manage basic credentials, covering the topic of user name and password management in general, and then focusing on the various FIM password management features at a high level.

Logon Names Versus Passwords



Logon

The primary job of FIM is the management of identity data across multiple systems, but it does more. We have seen how user attributes can be synchronized across systems—user names, logons, account names, or whatever they are called, can usually be synchronized like any other attribute.

Users may perceive a benefit in logons being consistent across systems, but clearly the issue of uniqueness in each system has to be addressed. A good way of addressing uniqueness is to establish one unique attribute for every user (and every new joiner) in the primary authoritative source(s). That attribute can then be used to generate account names, e-mail addresses, DNs, and so on.

If our only primary source were the FIM portal, we could perhaps make use of Account Name, which, as we have discussed, is checked for uniqueness in the portal. But our current configuration involves the Human Resources (HR) system as the primary authority for FTEs, as well as the portal being the primary authority for contractors. Clearly some agreed, overarching business process has to be identified, and then has to be implemented technically. This is generally the case in any organization, and every organization is different, so we cannot suggest just one right answer.

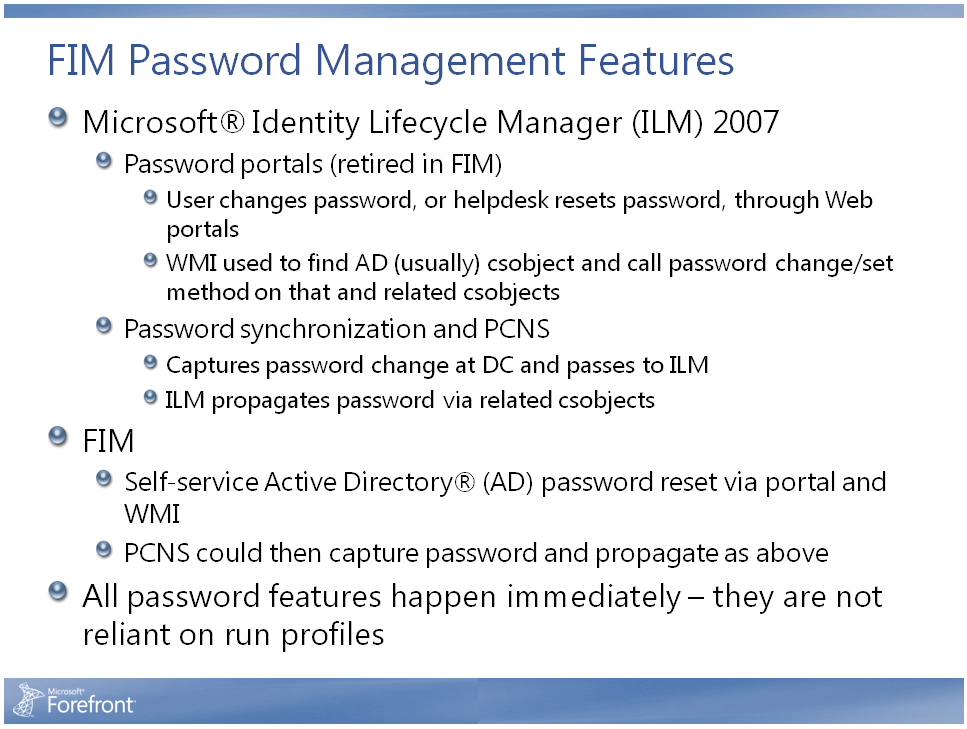
Passwords

Passwords cannot be synchronized in the same way. In a secure system it should not be possible to read a password at all and generally passwords are not stored, rather a tokenized or hashed version is stored.

Passwords can only be synchronized by FIM if they are captured at the moment they are changed, then FIM is capable of propagating passwords to a number of systems out-of-the-box, and can be extended (via code) to other systems. This immediate or event-based strategy is quite different from the usual

behavior of the synchronization service, which is batch-based. Notably, if password synchronization fails, it will not generally catch up at a later date. In order to get mismatched passwords back in sync, another password change may be required.

FIM Password Management Features



Password Portals

Right back to MIIS 2003, ILM has always shipped with two password portals, one for user password changes, and one for helpdesk password resets. These are implemented via extensible Web portals, the source code for which is provided. These are retired in FIM, but you need to know what they do for two reasons:

- A customer may still be using them (or a customer portal based on the same principles).
- The mechanism they use is still valid and to some degree used by the Password Change Notification Service (PCNS) and by FIM password reset workflows.

The password portals require that password changes (or helpdesk resets) are made through a portal, and not in the normal Windows® manner. The portals operate through the Synchronization Service using the Windows Management Instrumentation (WMI) interface. They ask for user name (sAMAccountName or userPrincipalName) and domain, and in the case of a change, also the old password. FIM uses these values to find the correct Active Directory® (AD) Connector Space (CS) object, and all related CS objects for the individual concerned, and for which the corresponding Management Agents (MAs) are configured for password operations. For those objects, it performs changes and/or resets in the connected systems. As configured, if the AD password change/reset fails, the process stops; if the AD one is successful, the others are tried, but if any of these fail, there is no concept of a retry (although the portal does report success or otherwise).

Password Synchronization and PCNS

From MIIS 2003 SP1, it has been possible to use PCNS to capture password changes at Domain Controllers (DCs) and pass them to the Synchronization Service, which sends them on to other connected systems for which it is configured.

It uses the same methods as the portals. Some Connected Data sources (CDs) can be handled out-of-the-box, but for others you need to write an extension. Unlike the portals, there is a concept of password queuing. FIM will retry a number of times (configurable), so if a server is temporarily unavailable there is more chance that the synchronization will complete.

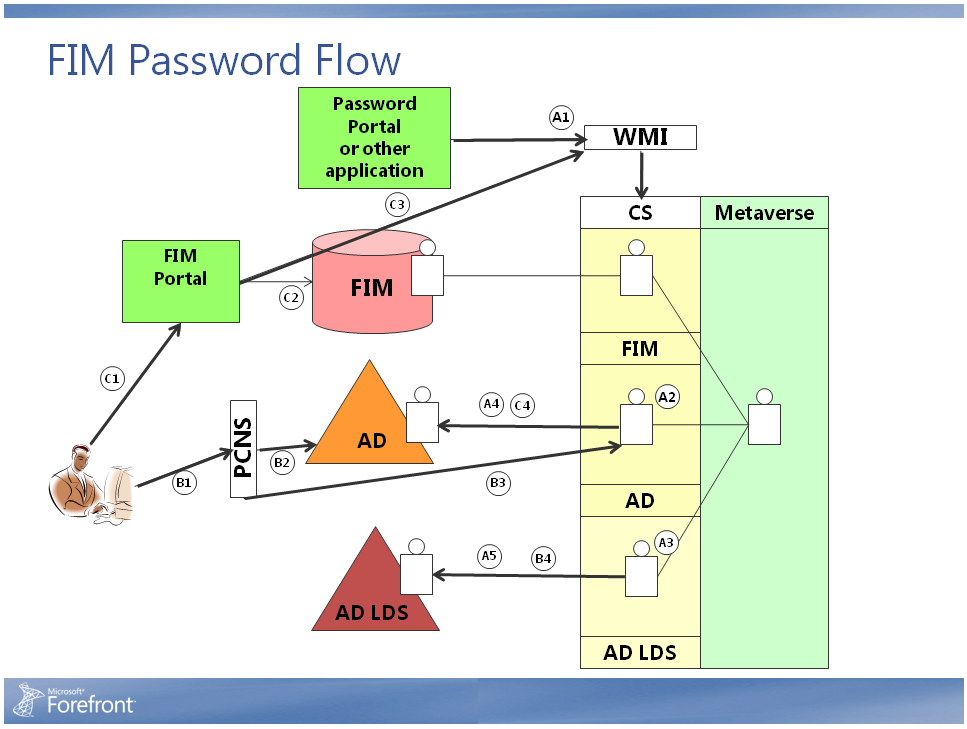
PCNS must be installed on every DC, although any configuration changes you need to make are propagated to all DCs.

FIM

FIM has all of the above capability (even though the old portals do not actually ship), plus self-service AD password reset via the portal (which uses WMI very much as before). PCNS could then capture the password changes and propagate as before, so that all passwords are kept in sync.

All password features happen immediately—they are not reliant on run profiles.

FIM Password Flow



WMI

The old ILM password portal is not provided as part of FIM, but the principle still applies (you could write your own portal):

A1. The portal passes the user name and domain (and in the case of a change, the old password) to FIM via the WMI.
A2. The correct AD CS object is found.
A3. Any other related objects, in MAs for which password management is configured, are found.
A4. A password change (user) or set (helpdesk) is sent to AD.
A5. If that is successful, a password set (never a change) is sent to any other CDs.

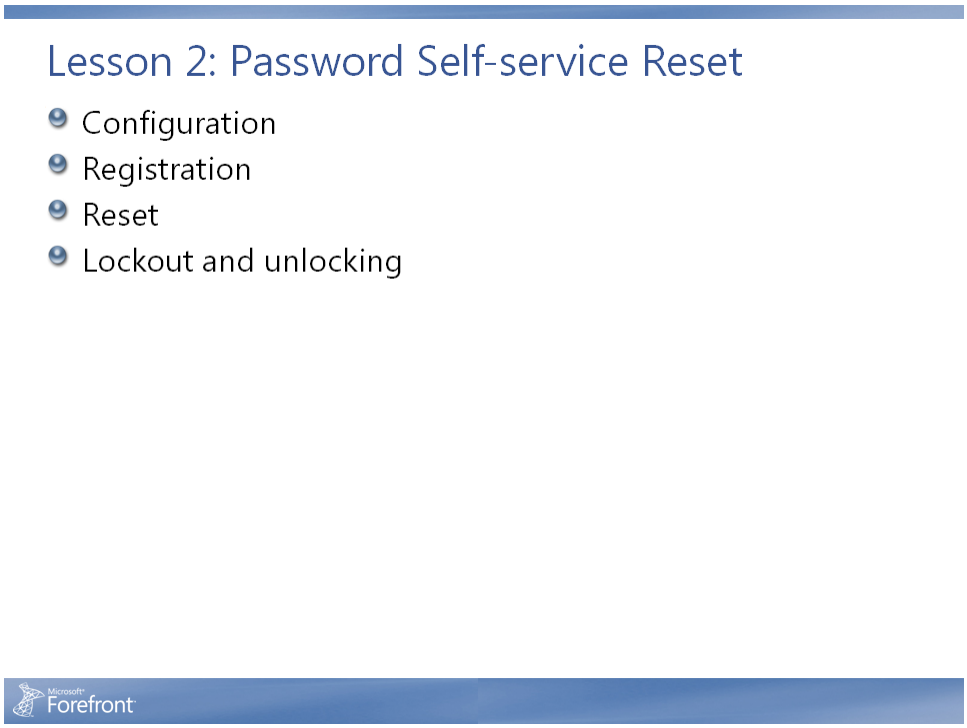
PCNS

B1. If a regular password change or set is made, and if PCNS is configured correctly on the DC, PCNS captures the password change or set.
B2. It is passed to the DC as usual.
B3. It is also passed to FIM (queuing on the DC if FIM is unavailable).
B4. FIM propagates the change to other systems for which password synchronization, and password management, has been configured.

Self-service Password Reset

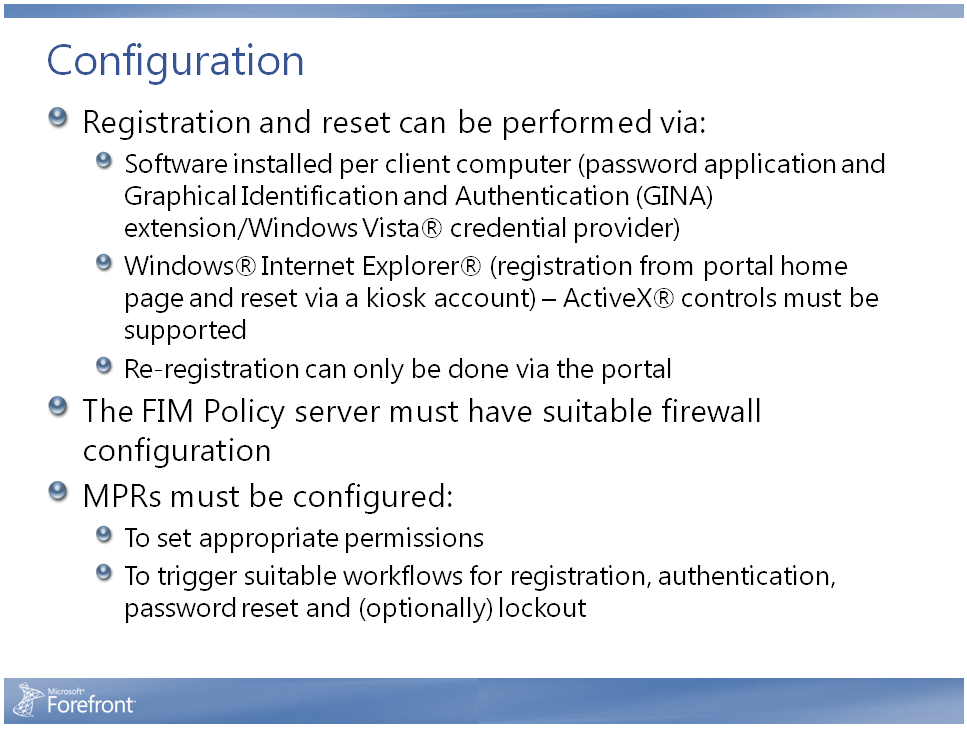
C1. A user performs a self-service password reset using the portal.
C2. The correct person is found (they are *anonymous* but provide a claimed account name, which must be the same as their sAMAccountName) and they are asked questions to which they must provide their secret answers.
C3. If they did that correctly, they are asked for a new password. This password reset is sent via WMI to...
C4. …AD.
C5. If PCNS is in use, this set is picked up by PCNS and propagated via FIM to other targets, as before.

Lesson 2: Password Self-service Reset



In this lesson we look at the new self-service password reset. In doing so, we must consider how it is configured and then how a user registers and later resets their password. If they (or someone else) tries and fails to provide the correct secret answers too many times, they can be locked out. We look at how this is configured and how they can get unlocked again.

Configuration



Password self-service registration and reset can be performed via either:

- Software installed on each client computer – A password reset application must be installed on clients, along with either a Windows® Graphical Identification And Authentication (GINA) extension, or using the Windows Vista®/Windows® 7 credential provider.
- Windows® Internet Explorer® – Registration is from the portal home page, and resets must be done via a kiosk account. Suitable security settings must be configured, including the support of ActiveX® controls.

If a user wishes to re-register or to change the answers to their questions (for example, because they have forgotten them), this can only be done via the portal.

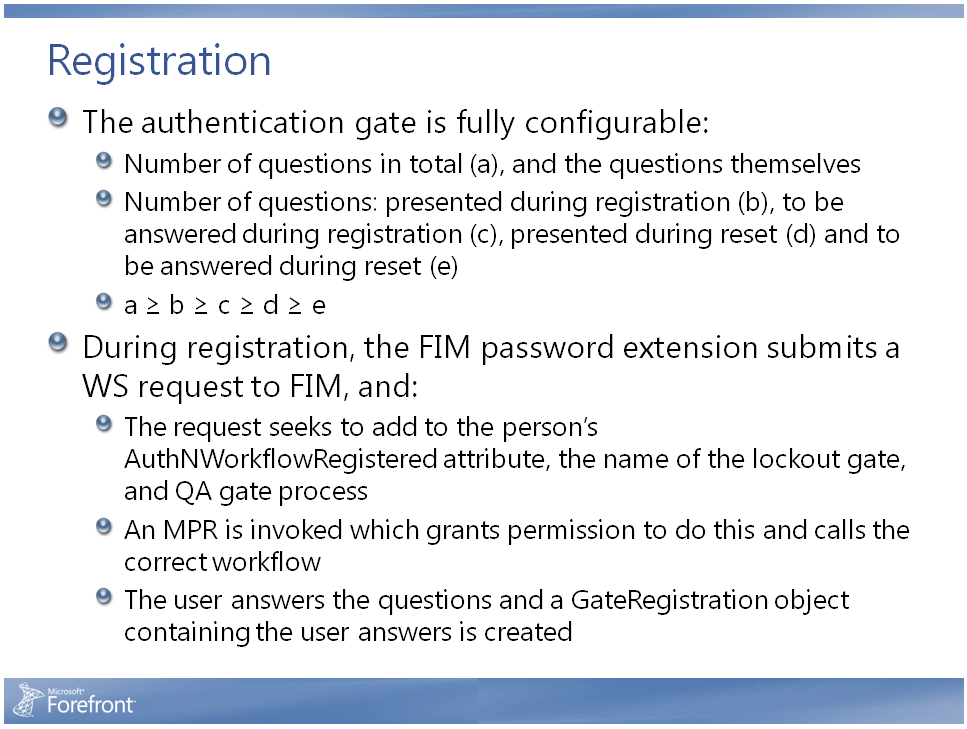
The FIM server must have suitable firewall configuration to allow all of this to take place. You will need the following inbound rules:

- Microsoft® Identity Management Common Service to allow inbound traffic through the local TCP port 526.
- Microsoft® Identity Management Security Token Service to allow inbound traffic through the local TCP port 527.
- Forefront Identity Manager Synchronization Service (RPC) to allow inbound traffic through Dynamic RPC.
- Forefront Identity Manager Synchronization Service (RPC-EPMAP) to allow inbound traffic through RPC Endpoint Mapper.

- Forefront Identity Manager Service (STS) (allows port 5726).
- Forefront Identity Manager Service (Web Service) (allows port 5725).

Management Policy Rules (MPRs) must be configured to grant appropriate permissions and to trigger suitable workflows for registration, authentication, password reset, and (optionally) lockout.

Registration



Authentication Gate

FIM ships with an authentication gate workflow. Of course, you could write a new one, but we are going to describe the out-of-the-box capabilities here.

The authentication gate is fully configurable for:

a. The number of questions in total, and the questions themselves.

b. The number of questions presented during registration.

c. The number of questions to be answered during registration.

d. The number of questions to be presented during reset.

e. The number of questions to be answered during reset.

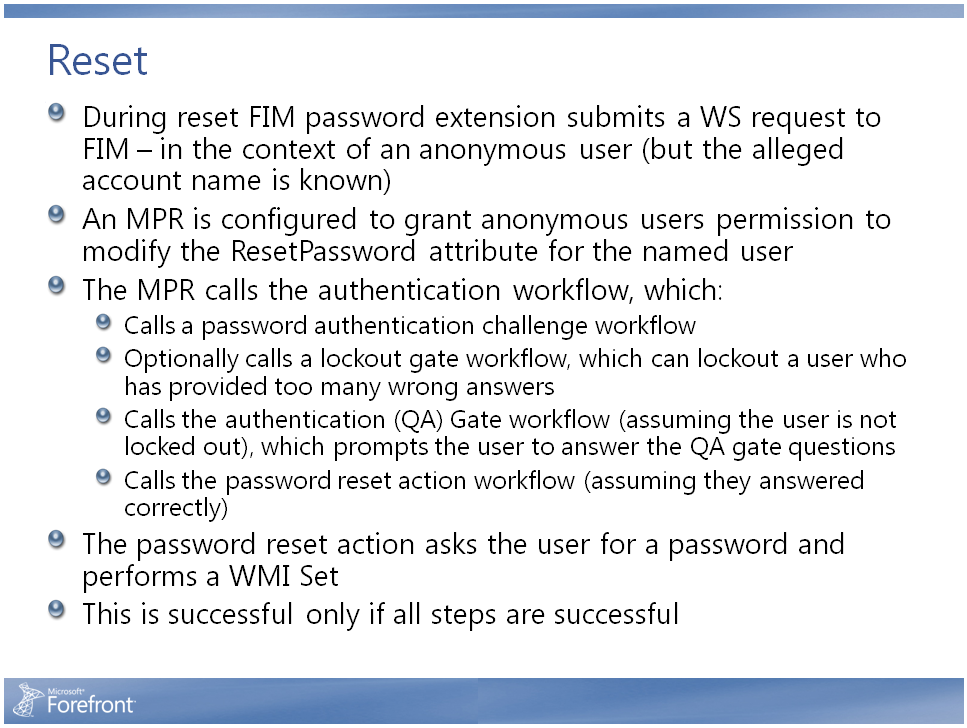
Note that a ≥ b ≥ c ≥ d ≥ e.

Registration

During registration, the FIM password extension submits a Web Service (WS) request to FIM, and:

- The request seeks to add to the person's AuthNWorkflowRegistered attribute, the name of the Lockout Gate and QA Gate process.
- This invokes an MPR which grants permission to do this and calls the /Person/AuthNWFRegistered workflow which presents the authentication gate questions.
- The user answers the questions and a GateRegistration object containing the (tokenized) user's answers is created.

Reset



When the user tries to perform a reset, the FIM password extension submits a WS request to FIM in the context of an anonymous user (but the alleged account name is known). An MPR (anonymous users can reset their password) is configured to grant anonymous users permission to modify the ResetPassword attribute for the named user.

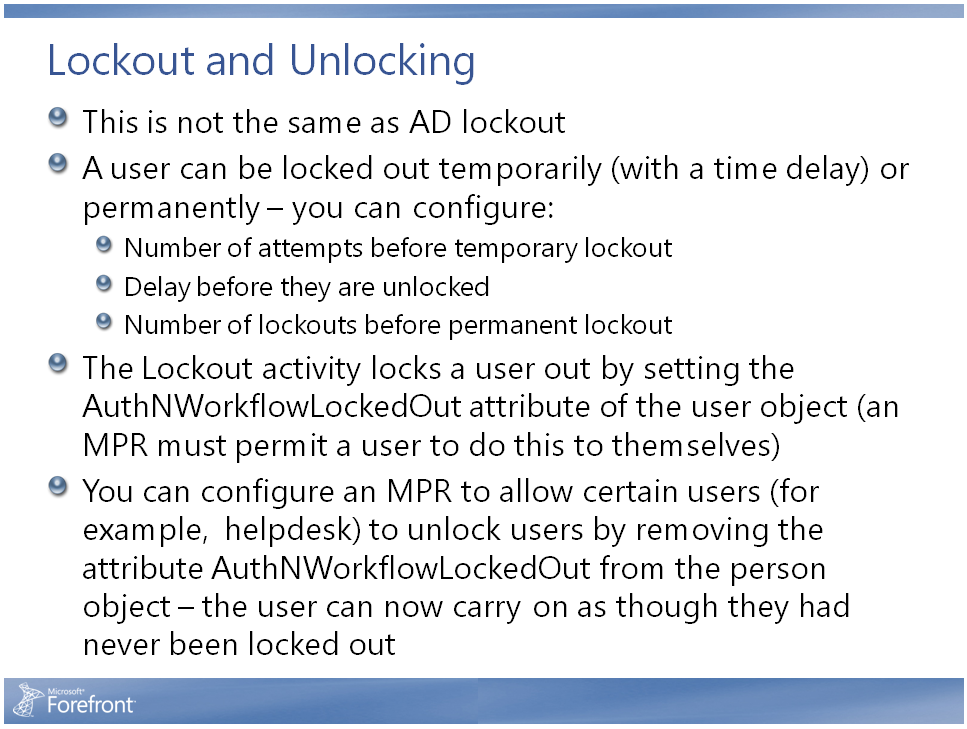
The MPR also calls the authentication workflow (Password Reset AuthN Workflow), which:

- Calls a Password Authentication Challenge workflow that is designed to stop someone walking up to a PC that has been carelessly left unlocked and unattended, and registering *on behalf* of someone else. This workflow is only used during registration.
- Optionally calls a lockout gate workflow (depending on configuration), which can lock out a user who has provided too many wrong answers. This workflow is only used during an attempt to reset.
- Calls the Authentication (QA) Gate workflow (assuming the user is not locked out), which prompts the user to ask or answer the QA gate questions (depending on whether the user is registering or attempting to reset).
- If they answered correctly, it runs the Password Reset Action workflow.

The Password Reset Action workflow asks the user for a password and then pushes that to AD (again via WMI).

This is successful only if all steps are successful, and even the last step can fail if, for example, the password doesn't meet policy requirements.

Lockout and Unlocking



Lockout is something handled by the FIM portal and it is not the same as an AD Lockout. When someone tries and fails to provide their secret answer correctly, and does this too many times (or indeed, when someone claiming to be them does so), we can arrange that they are locked out from performing a password reset, but this has no effect on their AD account (or any other account for that matter), unless of course, you configure that to happen (perhaps by using an MPR to flow a value to userAccountControl based on detecting a lockout).

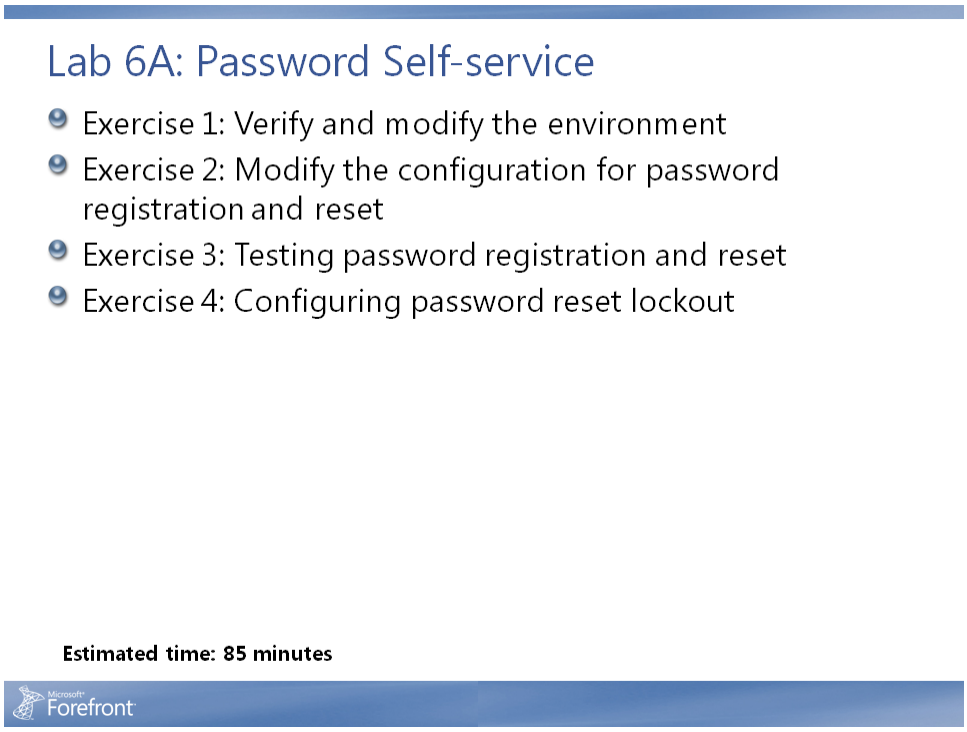
A user can be locked out temporarily, with a time delay until they are unlocked, or permanently. The usual set up is that you allow a number of temporary lockouts before a permanent lockout occurs. So in the out-of-the-box configuration, you can configure:

- The number of attempts before temporary lockout.
- The delay before they are unlocked.
- The number of lockouts before permanent lockout.

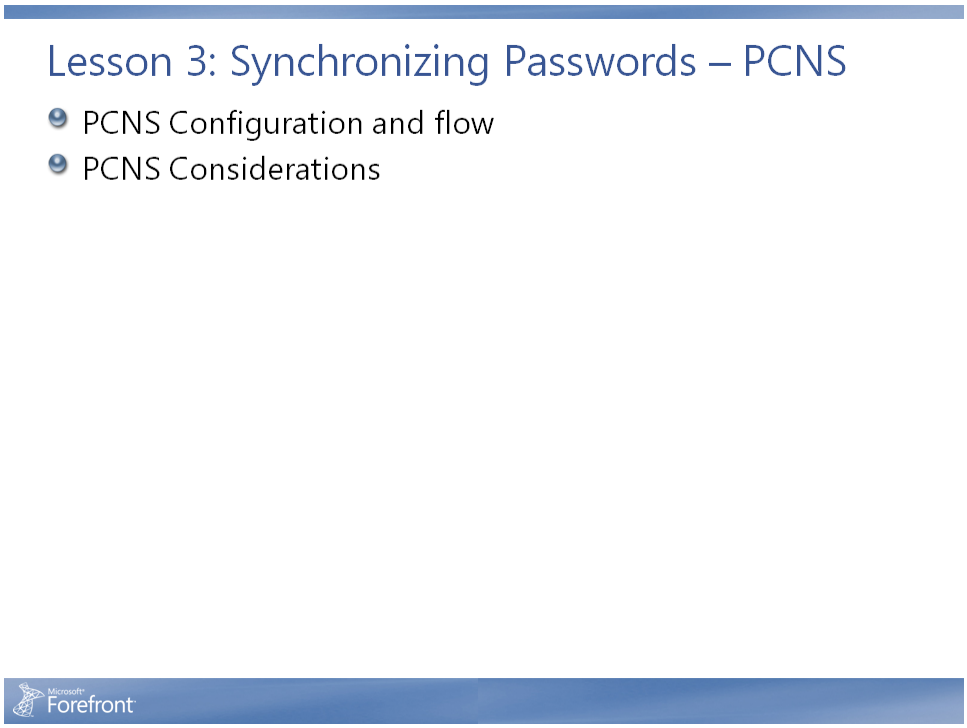
The lockout activity locks a user out by setting the AuthNWorkflowLockedOut attribute in the user object (an MPR must permit a user to do this to themselves).

You can configure an MPR to allow certain users (for example, helpdesk) to unlock users by resetting the attribute AuthNWorkflowLockedOut for the person object. The user can now carry on as though they had never been locked out.

Lab 6A: Password Self-service

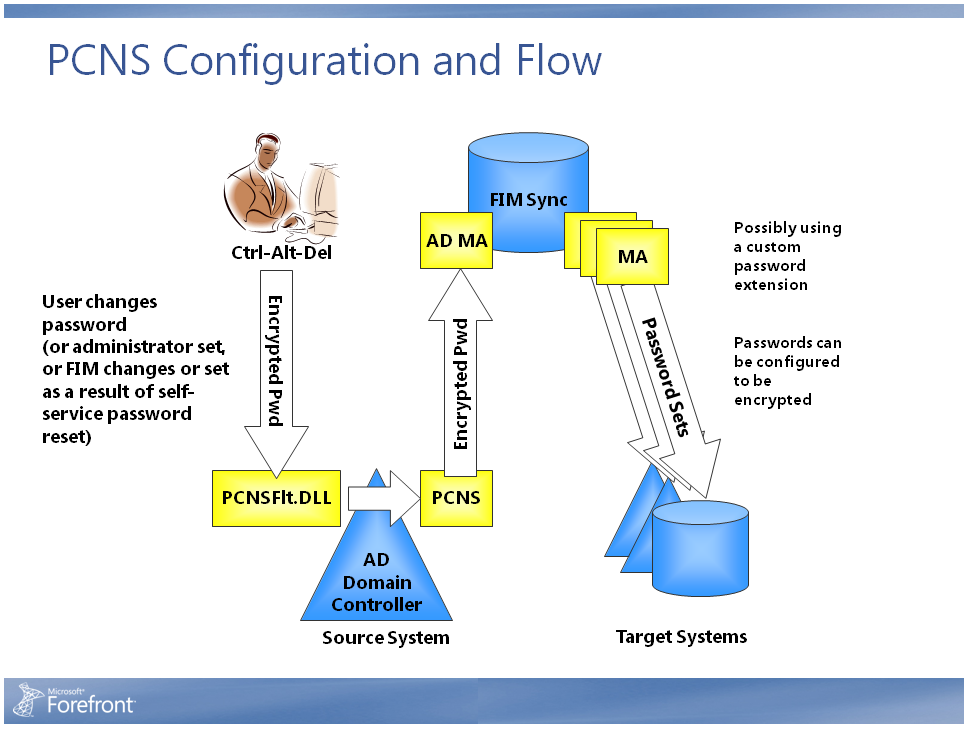


Lesson 3: Synchronizing Passwords – PCNS



This brief lesson is just a reminder of how PCNS works. The lab that follows contains the necessary configuration details.

PCNS Configuration and Flow

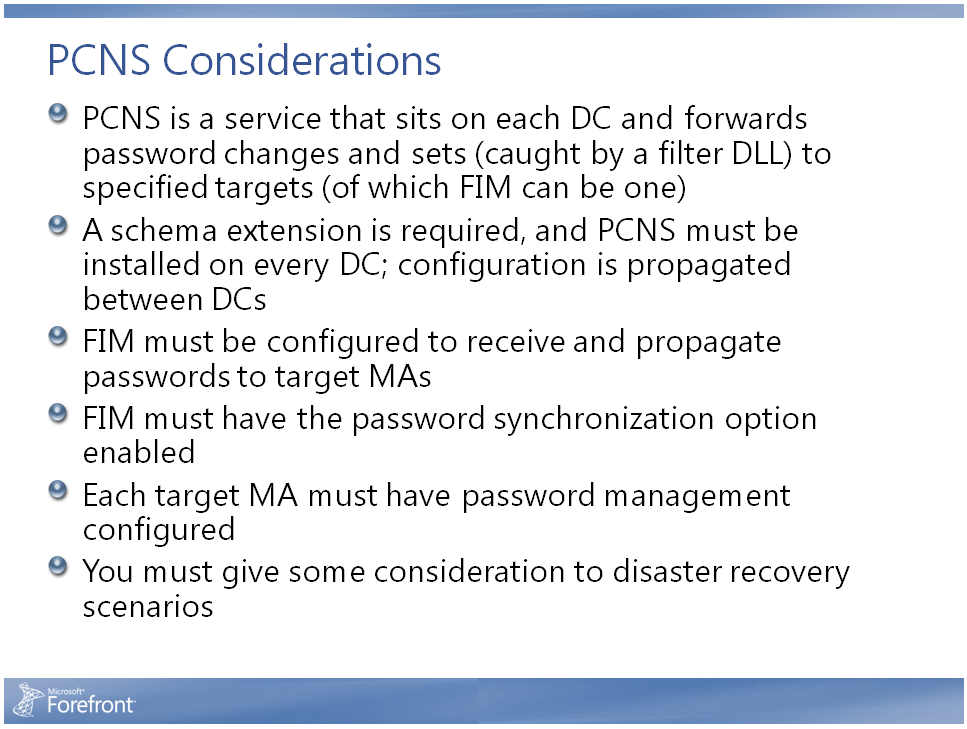


Password changes are captured by PCNS and the new password is sent securely to FIM, queuing up if necessary.

The receiving MA passes the new password on to other MAs. Most key directories are directly supported, if not, it is possible to write your own extensions. It would be normal to configure MAs to use Kerberos or Simple Authentication and Security Layer (SASL).

If a server cannot be contacted immediately, password changes will queue up in FIM for a limited time.

PCNS Considerations



PCNS is a service that resides on each DC and forwards password changes and sets (caught by a filter DLL) to specified targets (of which FIM can be one). So PCNS must be installed on all DCs and then configured to send changes to FIM. A schema extension is required, and configuration is propagated between DCs.

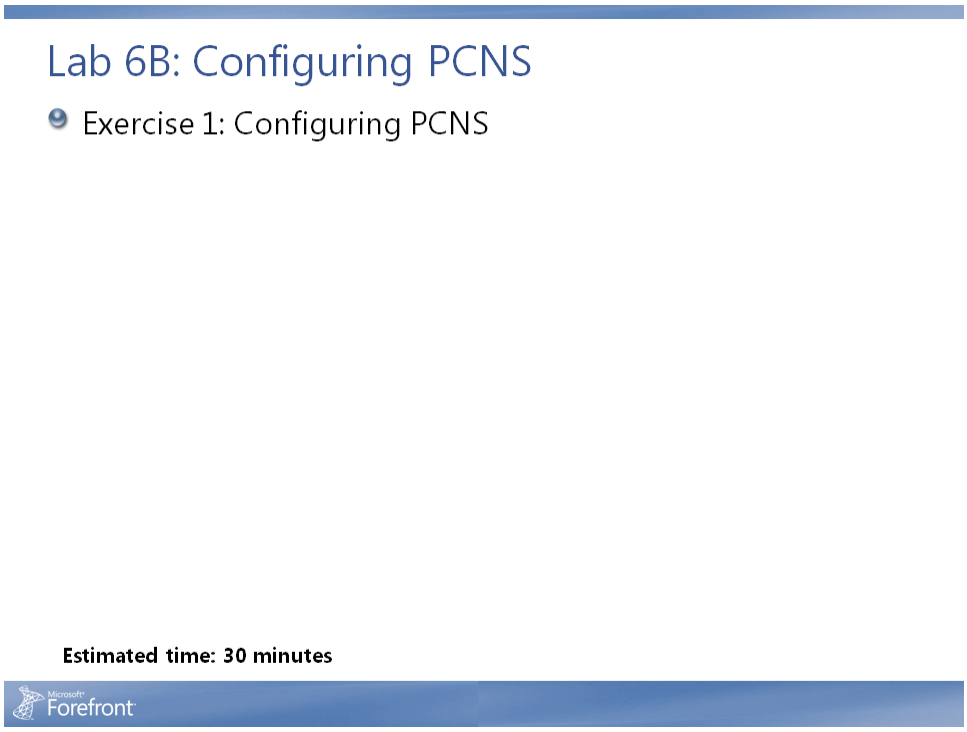
FIM must be configured to receive and propagate passwords from PCNS. The FIM MA for the domain(s) in question must be configured on a domain by domain basis to receive password changes from PCNS, and to propagate these changes to particular target MAs (for other forests, and other systems entirely).

In Synchronization Service Manager, the password synchronization option must be enabled.

For each target MA, password management must be configured (including how many retries, and how often, in the event that a system is unavailable).

You must give some consideration to disaster recovery scenarios. When you rebuild FIM, as long as the configuration is convergent, and you perform the rebuild correctly, you can expect to put things back as they were. However, passwords sit outside this paradigm. First, you will need to configure PCNS from the newly built FIM instance (if you have had to rebuild that), and second, any passwords unluckily in the pipeline when failure occurred will be lost. If users find inconsistencies in their password usage across systems, they will have to do another password change or reset.

Lab 6B: Configuring PCNS

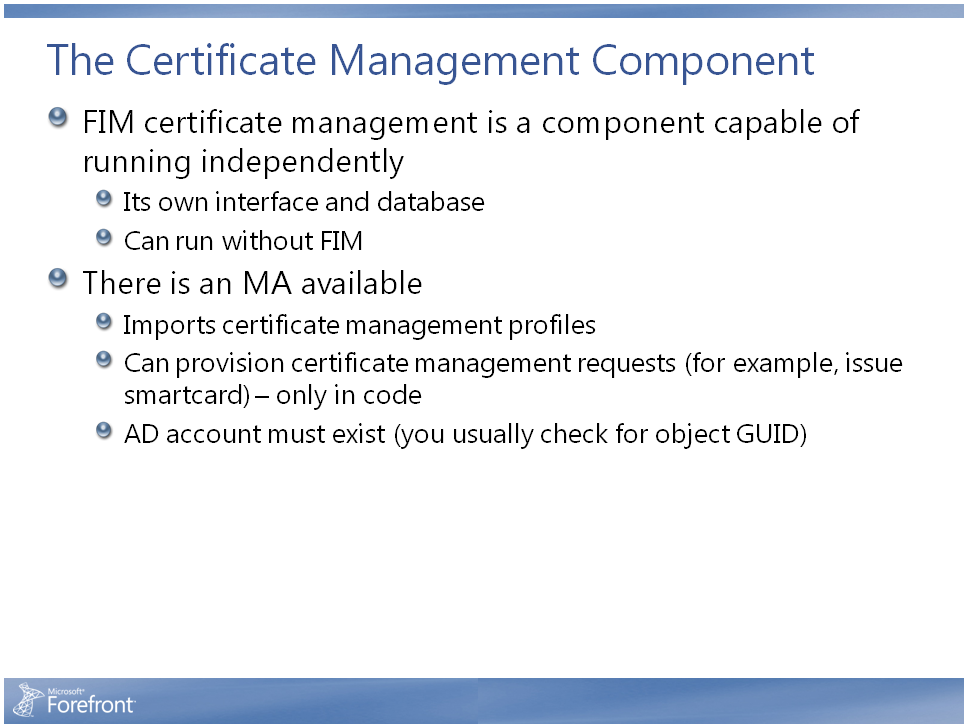


Lesson 4: FIM Certificate Management

Often when we talk of FIM, we mean the Synchronization Service, but FIM includes FIM Certificate Management (formerly called CLM).

In this brief lesson we simply consider the relationship between FIM Certificate Management and the other components of FIM, and describe how, in principle, you would provision requests for smartcard issue.

The Certificate Management Component



FIM Certificate Management as a Separate Component

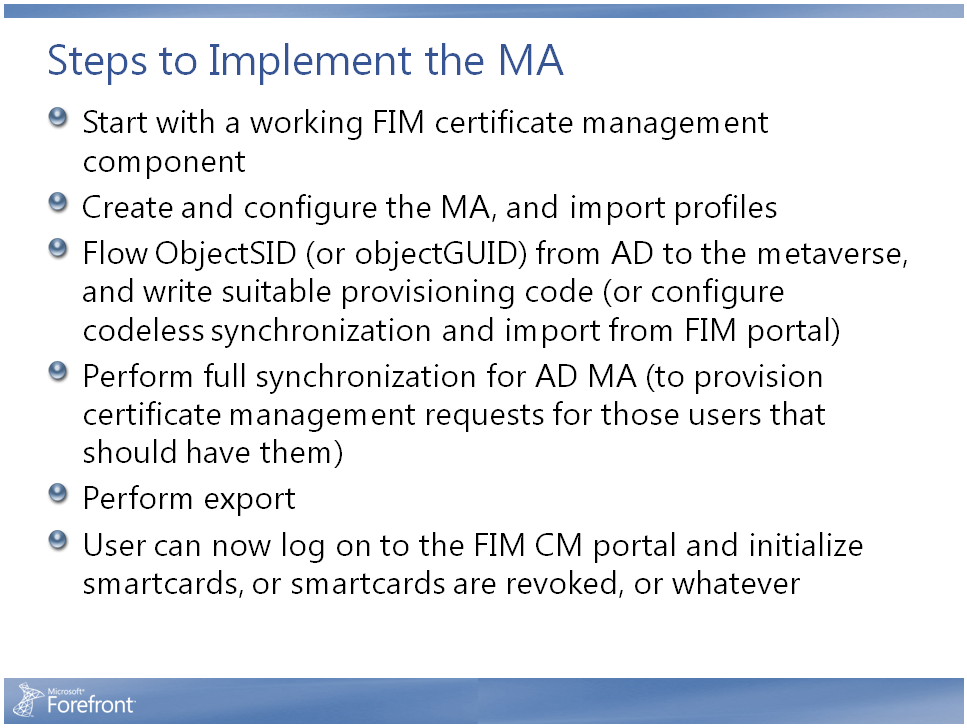
FIM Certificate Management is capable of running separately, with its own interface and database.

FIM Certificate Management as Part of FIM

The FIM Certificate Management MA imports Certificate Management profiles, and we can then provision requests to Certificate Management, rather like we provision any other object to any other system. So, for example, we can provision a Certificate Management request for smartcard issue, provided that the corresponding AD account exists. You would usually check for the existence of the AD objectGUID, which you flow into the MV.

This kind of provisioning will eventually be possible without code, but for the time being you need a provisioning rules extension.

Steps to Implement the MA



You need to start with a working Certificate Management system. It must be possible to use FIM Certificate Management to provision certificates on smartcards (if that is what you want to do via FIM).

There is a Certificate Management MA, which you create like any other and use to import the profiles from FIM Certificate Management.

Since you cannot provision a Certificate Management request without there being a corresponding AD account, you must arrange that objectSID or objectGUID flows into the MV so that the code can check for existence (you would flow objectSID in any way to support portal log on).

Finally, and this is the difficult part, you must write suitable provisioning code, then perform full synchronization for the AD MA (to provision requests in Certificate Management for those users that should have them), and an export to Certificate Management.

Users can now log on to Certificate Management and initialize smartcards, and if you provision other requests, you can revoke the cards.

For more information, see **ILM 2007 Getting Started Collection** (http://www.microsoft.com/downloads/details.aspx?FamilyID=11FB01BC-94A9-4404-BB90-CECA1A206E32&displaylang=en).