

Imanami Corporation

5099 Preston Ave,
Livermore, CA 94551

Recipient Update Service Installation & Configuration Guide

For Microsoft Exchange 2007

Author: Robert Haaverson

Copyright 2007 Imanami Corporation. All Rights Reserved.

Recipient Update Service Installation & Configuration Guide

For Microsoft Exchange 2007

Overview

Microsoft Exchange 2007 handles recipient management different than the previous versions. In Exchange 2000 and 2003, the Recipient Update Service (RUS) was responsible for adding new recipients (mailbox-enabled users, mail-enabled users, mail-enabled contacts, and mail-enabled groups) to address lists and ensuring Recipient Policies were applied to the new recipient. In Exchange 2007, the RUS is noticeably absent and you must call the Exchange API's to ensure a recipient is provisioned correctly.

Calling new API's requires that Imanami's existing recipient provisioning routines be re-written to call Exchange's API which is implemented as PowerShell Cmdlets (pronounced command-lets). All new versions of Imanami products support this API but they are not scheduled to release until 2008. In the meantime many Imanami customers (old and new) require Exchange 2007 support in the existing release of the Imanami products.

How can Imanami add support for Exchange 2007 to the existing code base without modifying the shipped code and requiring a lengthy regression test for each product?

Answer: Create a service that runs on one machine that corrects any improperly provisioned Exchange 2007 objects in Active Directory.

Environments

The Imanami RUS for Exchange 2007 supports the following environments:

1. Exchange 2007 installed in a mixed environment with Exchange 2003 and the Exchange 2003 Recipient Update Service is present and running.
2. Exchange 2007 installed in a mixed environment with Exchange 2003 and the Exchange 2003 Recipient Update Service is present but **NOT** running.
3. Exchange 2007 installed in a native environment without any legacy Exchange servers. No Exchange 2003 Recipient Update Service is present.

Architecture

The Recipient Update Service for Exchange 2007 (RUS2007) consists of three components:

1. PowerShell Script (PS1) – Performs all actions
2. Windows Service (C# EXE) – Hosts the script
3. Class Library (C# DLL) – Provides heavy lifting to #1 above

The Windows Service runs the PowerShell Script at a pre-determined interval (every 60 seconds). The PowerShell Script is where the real logic resides. The PowerShell script uses the Class Library when exceptional heavy lifting is required.

Functionality

Program flow:

1. Autodetect the forest and domains. This setting is user configurable. The user can specify which forest or domain to process but the account the service runs under must have permissions within that forest or domain.
2. Is Exchange 2007 installed in the forest? If yes then go to 3. If no then End and notify the user.
3. Are the Exchange 2007 Administration Tools installed? If yes then go to 4. If no then End and notify the user. We need the admin tools to fix the recipients.
4. For each domain discovered:
 - a. Find all improperly provisioned recipients (mailbox-enabled users, mail-enabled users, mail-enabled contacts and mail-enabled groups).
Note: If the Exchange 2003 RUS is present the script will only process mailbox-enabled users, it will not process mail-enabled recipients such as mail-enabled users, contacts and group. The Exchange 2003 RUS will process those recipients correctly. See the Exchange Team blog for administration co-existence:
<http://msexchangeteam.com/archive/2006/10/09/429135.aspx>

i. Mailbox-Enabled Users

```
"(&(objectClass=User)(objectCategory=Person)(mailNickname=*)(!msExchVersion=*)(homeMDB=$databaseDN))"
```

Note: *The homeMDB=\$databaseDN predicate ensures only mailboxes on the specified Exchange 2007 database are returned.*

ii. Mail-Enabled Users

```
"(&(objectClass=User)(objectCategory=Person)(mailNickname=*)(!msExchVersion=*)(targetaddress=*)(!msExchALObjectVersion=*))"
```

Note: *The !msExchangeALObjectVersion=* predicate prevents the query from returning objects that have already been processed by the Exchange 2003 RUS.*

iii. Mail-Enabled Contacts

```
"(&(objectClass=Contact)(objectCategory=Person)(mailNickname=*)(!msExchVersion=*)(targetaddress=*)(!msExchALObjectVersion=*))"
```

Note: *The !msExchangeALObjectVersion=* predicate prevents the query from returning objects that have already been processed by the Exchange 2003 RUS.*

iv. Mail-Enabled Groups

```
"(&(objectClass=Group)(objectCategory=Group)(mailNickname=*)(!msExchVersion=*)(displayname=*)(!mail=*)(!msExchALObjectVersion=*))"
```

Note: *The !msExchangeALObjectVersion=* predicate prevents the query from returning objects that have already been processed by the Exchange 2003 RUS.*

b. For each recipient from 4a:

i. Fix the recipient.

Note: *we cannot call an Exchange 2007 cmdlet to fix the recipient (such as Set-Mailbox – ApplyMandatoryParameters) because Exchange 2007 will not recognize the recipient created by Imanami products.*

1. Verify the recipient needs fixing (IsValidMailbox(), IsValidMailUser(), etc.).
2. Save the following attribute values which are required to re-provision the recipient in step 4.b.i.4:

a. Mailbox-Enabled Users

- DistinguishedName for the Identity parameter.
- MailNickName for the Alias parameter.

- HomeMDB for the Database parameter.
- b. Mail-Enabled Users
- DistinguishedName for the Identity parameter.
 - MailNickName for the Alias parameter.
 - Targetaddress for the ExternalEmailAddress parameter.
- c. Mail-Enabled Contacts
- DistinguishedName for the Identity parameter.
 - MailNickName for the Alias parameter.
 - Targetaddress for the ExternalEmailAddress parameter.
- d. Mail-Enabled Groups
- DistinguishedName for the Identity parameter.
 - MailNickName for the Alias parameter.
 - DisplayName for the DisplayName parameter.
3. Strip off all Exchange related attributes. Required for step 4bi4.
- mailnickname
 - homemdb
 - homemta
 - mDBUseDefaults
 - mail
 - proxyAddresses
 - msexchversion
 - msExchALObjectVersion
 - legacyExchangeDN

- msExchMailboxGuid
 - msExchMailboxSecurityDescriptor
 - msExchPoliciesIncluded
 - msExchRecipientDisplayType
 - targetAddress
 - mAPIRecipient
 - internetEncoding
4. Call the appropriate Exchange 2007 cmdlet to provision the object correctly (Enable-Mailbox, Enable-MailUser, etc.).
 - a. Mailbox-Enabled Users
 Enable-Mailbox -Identity '\$identity' -Alias '\$alias' -Database '\$homeMDB'
 - b. Mail-Enabled Users
 Enable-MailUser -Identity "\$identity" -Alias "\$alias" -ExternalEmailAddress "\$targetaddress"
 - c. Mail-Enabled Contacts
 Enable-MailContact -Identity \$identity -Alias \$alias -ExternalEmailAddress \$targetaddress
 - d. Mail-Enabled Groups
 Enable-DistributionGroup -Identity "\$identity" -Alias "\$alias" -DisplayName "\$display"
 - ii. Write to the event log which recipient was fixed and how it was fixed (which attributes were removed and which command was called).
5. Wait 10 seconds. This setting is user configurable. You could set it for 10 minutes if you want.
 6. Go to step 1 and start all over, you never know when a new recipient will show up.

Installation

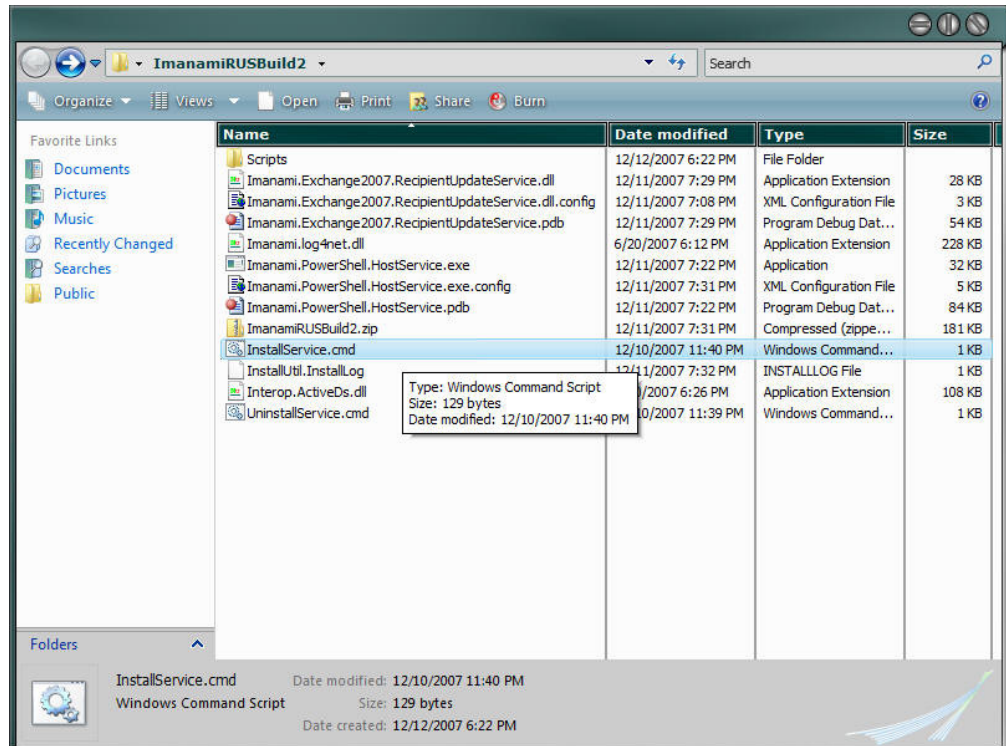
The Imanami Recipient Update Service for Microsoft Exchange 2007 consists of eight files:

- Imanami.Exchange2007.RecipientUpdateService.dll
- Imanami.Exchange2007.RecipientUpdateService.dll.config
- Imanami.PowerShell.HostService.exe

- Imanami.PowerShell.HostService.exe.config
- Scripts\ImanamiRUS.ps1
- Scripts\AddService.ps1
- Scripts\RemoveService.ps1
- InstallService.cmd
- UninstallService.cmd
- Interop.ActiveDs.dll
- Imanami.log4net.dll

Installing the service is as simple as:

1. Install Prerequisites:
 - a. Install Windows PowerShell.
 - b. Install Microsoft Exchange 2007 Administration Tools.
 - c. Modify the default execution policy of Windows PowerShell to RemoteSigned:
 - i. Set-ExecutionPolicy RemoteSigned
2. Extract the files from the zip file into your favorite folder such as c:\Program Files\Imanami\RUS2007.
3. Double click on the InstallService.cmd file.

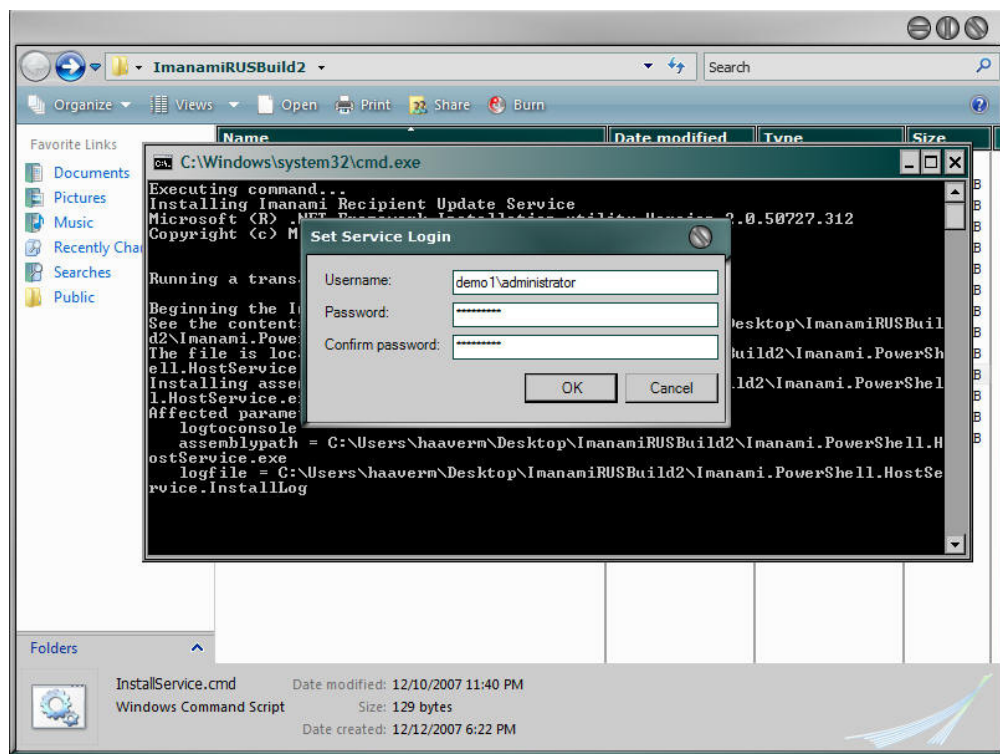


4. Enter the domain\username and password for the account the service should run under. This account needs permission Exchange Admin permission in Exchange 2007. It will be used to call Enable-Mailbox, Enable-MailUser, Enable-MailContact, and Enable-Group when it finds a recipient requiring repair.
 Note: By default, RUS2007 scans every domain in the forest that the username specified in step 3 resides. To connect to a different forest specify a domain\username of a user in the target forest in step 3.

5. The service is automatically started after you enter the logon information. The service is set to start automatically when the computer boots up.

Configuration

The following settings are configurable by modifying the Imanami.PowerShell.HostService.exe.config file in Notepad.



Setting	Description
ServiceScriptArguments	<p>Arguments to pass to the RUS. The RUS supports 3 parameters:</p> <ul style="list-style-type: none"> -Domains – comma separated list of domains in the forest to process. Default is to process all domains in the forest. The RUS only processes the domains in the same forest as

	<p>the domain specified for the service.</p> <p>-LoopInterval – number of seconds to sleep between processing. Default is 60 seconds in the configuration file. If this parameter is not specified then the default behavior is to run once and exit (do not loop).</p> <p>-FixErrors – Whether to fix broken recipients. Default is true in the configuration file. If this parameter is not specified then the default is false.</p>
ServiceScript	Script filename for the service. Do not specify the path. The service will look first in the Scripts folder and then in the folder in which the service executable resides. Changing this setting is not supported.
ServiceIntervalMilliseconds	Number of milliseconds to wait between checking on the script (to ensure it is running). Changing this setting is not supported.
ServiceName	Name of the service as displayed in the Services MMC. Changing this setting is not supported.
ServiceDescription	Description of the service as displayed in the Services MMC. Changing this setting is not supported.
RequiredSnapIns	PowerShell snapins required by the service script. Changing this setting is not supported.
RequiredAssemblies	Assemblies required by the service script. Changing this setting is not supported.

There are other settings in the file, but the end user need only be concerned with the above settings.

Default Imanami.PowerShell.HostService.exe.config file:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="log4net"
type="System.Configuration.IgnoreSectionHandler" />
    <sectionGroup name="applicationSettings"
type="System.Configuration.ApplicationSettingsGroup, System,
Version=2.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089" >
      <section
name="Imanami.PowerShell.HostService.Properties.Settings"
type="System.Configuration.ClientSettingsSection, System,
Version=2.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089" requirePermission="false" />
    </sectionGroup>
  </configSections>
  <applicationSettings>
    <Imanami.PowerShell.HostService.Properties.Settings>
```

```

        <setting name="ServiceScript"
serializeAs="String">
            <value>ImanamiRUS.ps1</value>
        </setting>
        <setting name="ServiceIntervalMilliseconds"
serializeAs="String">
            <value>30000</value>
        </setting>
        <setting name="ServiceName" serializeAs="String">
            <value>Imanami Recipient Update
Service</value>
        </setting>
        <setting name="ServiceDescription"
serializeAs="String">
            <value>Imanami Recipient Update Service for
Exchange 2007</value>
        </setting>
        <setting name="RequiredSnapIns" serializeAs="Xml">
            <value>
                <ArrayOfString
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
                    <string>Microsoft.Exchange.Management.PowerShell.Admin</string>
                </ArrayOfString>
            </value>
        </setting>
        <setting name="RequiredAssemblies"
serializeAs="Xml">
            <value>
                <ArrayOfString
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
                    <string>Imanami.Exchange2007.RecipientUpdateService.dll</string>
                </ArrayOfString>
            </value>
        </setting>
        <setting name="ServiceScriptArguments"
serializeAs="String">
            <value>-LoopInterval 60 -
FixErrors:$true</value>
        </setting>
    </Imanami.PowerShell.HostService.Properties.Settings>
</applicationSettings>
<!-- This section contains the log4net configuration
settings -->
<log4net>
    <!-- Define some output appenders -->
    <appender name="RollingLogFileAppender"
type="log4net.Appender.RollingFileAppender">

```

```

        <file value="$ {TMP} \~PowerShellHostService-log.txt"
/>
        <appendToFile value="false" />
        <rollingStyle value="Once" />
        <datePattern value="yyyyMMdd-HHmm" />
        <maxSizeRollBackups value="10" />
        <maximumFileSize value="100MB" />
        <staticLogFileName value="true" />
        <layout type="log4net.Layout.PatternLayout">
            <header value="[PowerShell.HostService Logfile
Start]&#13;&#10;" />
            <footer value="[PowerShell.HostService Logfile
End]&#13;&#10;" />
            <conversionPattern value="%date [%thread] %-5level
%logger [%property{NDC}] - %message%newline" />
        </layout>
    </appender>
    <appender name="DetailedRollingLogFileAppender"
type="log4net.Appender.RollingFileAppender">
        <file value="$ {TMP} \~PowerShellHostService-log.txt"
/>
        <appendToFile value="false" />
        <rollingStyle value="Once" />
        <datePattern value="yyyyMMdd-HHmm" />
        <maxSizeRollBackups value="10" />
        <maximumFileSize value="100MB" />
        <staticLogFileName value="true" />
        <layout type="log4net.Layout.PatternLayout">
            <header value="[PowerShell.HostService Logfile
Start]&#13;&#10;" />
            <footer value="[PowerShell.HostService Logfile
End]&#13;&#10;" />
            <conversionPattern value="%date{HH:mm:ss,fff}
[%thread] %-5level %type{1}:%M() - %message%newline" />
        </layout>
    </appender>
    <!-- Setup the root category, add the appenders and set
the default level -->
    <root>
        <level value="ERROR" />
        <appender-ref ref="DetailedRollingLogFileAppender" />
    </root>
</log4net>

</configuration>

```

Appendix A – How Imanami Products Provision Recipients for Exchange 2003

Imanami products, in their current version (circa 2007), depend on the Exchange 2003 Recipient Update Service to provision Exchange 2003 recipients. The Exchange 2003 Recipient Update Service does not provision Exchange 2007 recipients correctly. This section covers how Imanami products “seed” recipients for the Exchange 2003 Recipient Update Service. This information is required so the RUS2007 knows what an Imanami provisioned object looks like in the directory and can differentiate between a corrupt Exchange 2007 recipient created by Imanami products versus a corrupt Exchange 2007 created by another product like the Exchange 2003 ADUC.

Imanami Directory Transformation Service (DTM)

The current version of Imanami DTM (version 3.0) seeds recipients by providing the following attributes:

1. Mailbox-enabled User
 - a. MailNickName
 - b. HomeMDB
 - c. DisplayName
2. Mail-enabled User
 - a. MailNickName
 - b. TargetAddress
 - c. DisplayName
3. Mail-enabled Contact
 - a. MailNickName
 - b. TargetAddress
 - c. DisplayName
4. Mail-enabled Group
 - a. MailNickName
 - b. DisplayName

Note that the user can add additional attributes through the DTM console. DTM 4.0 will require more attributes to be set for Exchange 2003 support. See Appendix B – Required Attributes for the Exchange 2003 Recipient Update Service.

Imanami Directory SmartDL

The current version of Imanami SmartDL (version 4.0) seeds groups by providing the following attributes:

1. Mail-enabled Group
 - a. MailNickName
 - b. DisplayName
 - c. LegacyExchangeDN

SmartDL 5.0 will set all the required attributes for Exchange 2003 support. See Appendix B – Required Attributes for the Exchange 2003 Recipient Update Service.

Imanami Directory SmartR

SmartR version 2.0 does not create Exchange recipients.

Imanami Directory WebDir

The current version of Imanami WebDir (version 4.0) seeds recipients by providing the following attributes:

1. Mailbox-enabled User
 - a. MailNickName
 - b. HomeMDB
 - c. DisplayName
2. Mail-enabled User
 - a. MailNickName
 - b. TargetAddress
 - c. DisplayName
3. Mail-enabled Contact
 - a. MailNickName
 - b. TargetAddress
 - c. DisplayName
4. Mail-enabled Group
 - a. MailNickName
 - b. DisplayName