App2Me V1.5
Security White Paper

Revision 1.00

Revision History

Revision	Date	Summary	Author
1.00	2010/1/22	New	
Revisions			Revision Reason

Contents

1. Introduction

This document explains the data exchange and storage methods used by App2Me. This information is provided for security purposes.

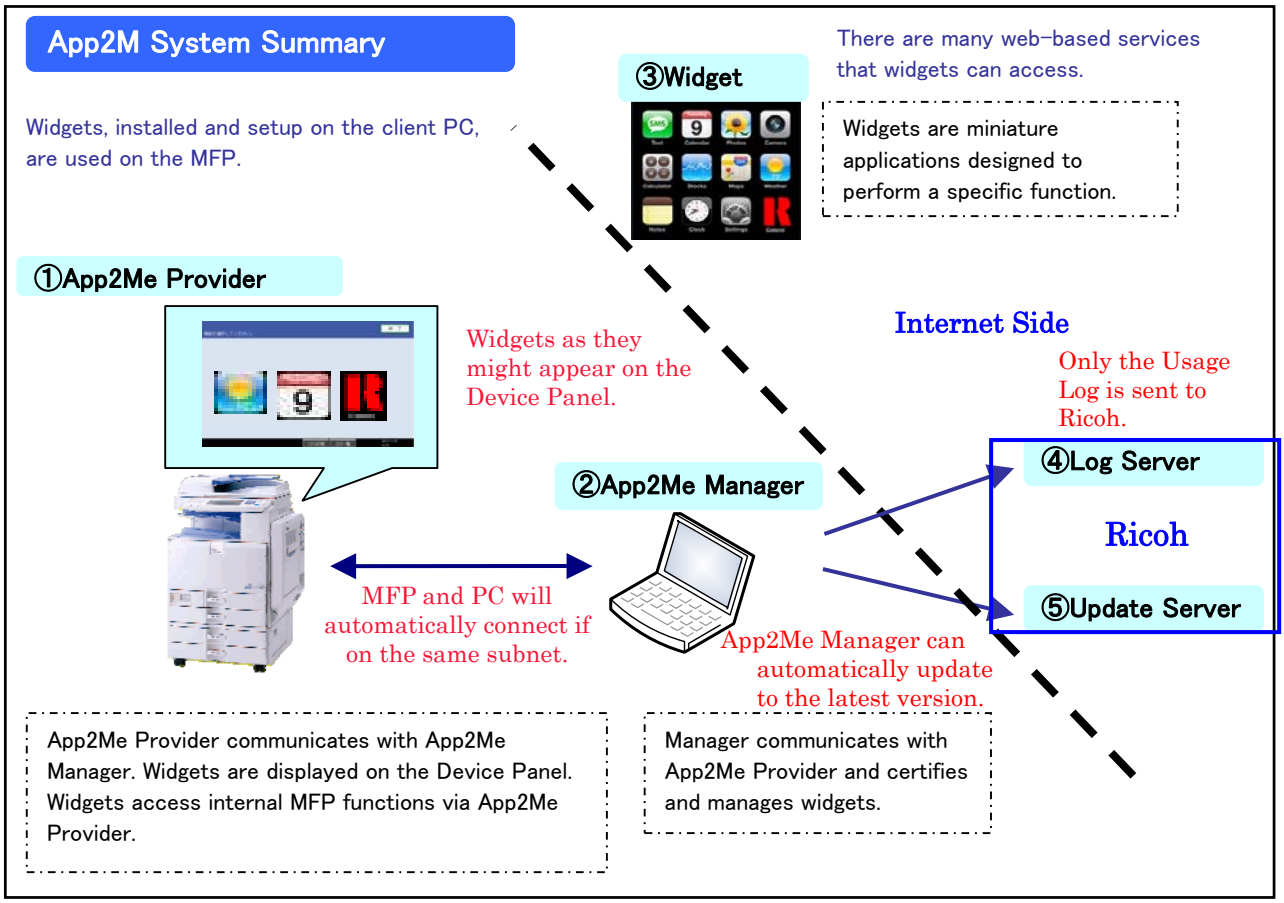
The following items are covered by this document:

- Network Protocols
- Data Flow
- Data Storage
- Security Concerns

2. App2Me Configuration

This is a quick overview of the components used by App2Me and how those components are connected. Please refer to the App2Me Startup Guide for details. The guide is available from the App2Me official website.

2.1. Diagram



2.2. Components List

Component	Description
App2Me Provider	App2Me Provider is the SDK Application that runs on the MFP. It gives users access to their widgets on the MFP, and gives the widgets access to device functions.
App2Me Manager	App2Me Manager is the application that runs on the client PC. It manages the Widgets, user information, and communicates with App2Me Provider.
Widget	Each widget is an application with a single or limited function that is used with the selected Widget Engine (for example, Google Desktop). Widgets access the scanning and printing functions on the MFP and synchronize with other applications, such as Web Services, to add value and functionality to a MFP.
Log Server	App2Me Manager automatically transfers the Usage Log to this server if the Usage Log is being collected (it is on by default).
Update Server	If App2Me Manager is set to automatically update, or a user manually selects to update, a connection is made to this server to look for and download any available updates.

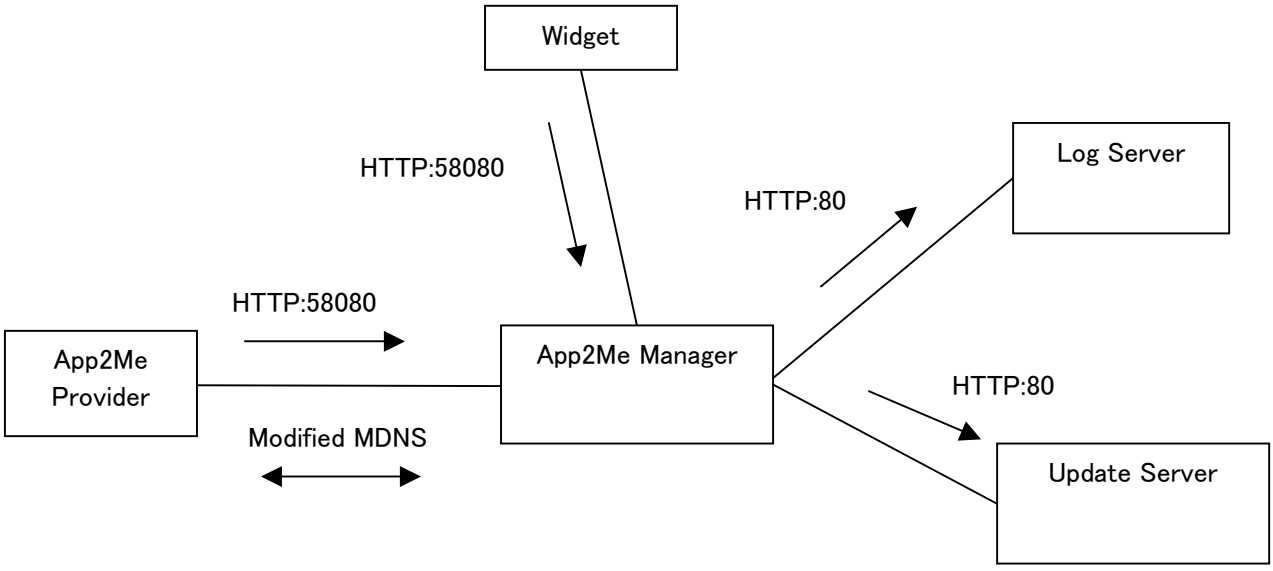
3. Network Protocol

Explanation of the Network Protocols that App2Me utilizes.

3.1. List of Protocol Ports

Component	Protocol	Application Protocol	Port Number	Remark
App2Me Manager	TCP	http	58080	Both Widgets and App2Me Provider uses this protocol to access the App2Me Manager. This port cannot be changed.
	UDP	Modified MDNS (*1)	Variable 1024-65535	Default Port Number is 55353. This protocol is used by App2Me Manager to discover MFPs. This port number is variable, but 5353 is not allowed as the standard MDNS port is 5353. For more information see the operation chapter.
App2Me Provider	UDP	Modified MDNS (*1)	Variable 1024-65535	Default Port Number is 55353. This protocol is used by App2Me Provider to discover App2Me Manager. This port number is variable, but 5353 is not allowed as the standard MDNS port is 5353. For more information see the FAQs section. (Chapter 8)
Update Server	TCP	http	80	
Log Server	TCP	http	80	

*1: This is a variant of MDNS produced by Ricoh, and is based on the draft RFC. See the FAQ section for more information.



4. Data Flow and Data Storage

The data that App2Me handles can be divided into the following two categories:

- Data exchanged between the various App2Me components.
- Data that is saved and stored by the various App2Me components.

4.1. Data Flow

(1) Between Widget and App2Me Manager (HTTP)

In order to use the print and scan functions, Widgets must communicate with the App2Me Manager. The following data is exchanged:

Data	Originator/Sender	Control/Request	Flow
Widget Information ・ Widget Name ・ Widget PIN ・ Widget Icon	Widget	Add Delete	Registers/deletes Widget Information with App2Me Manager.
Printing Job ・ File to Print ・ Printing parameters	Widget	Add Delete Change	Registers/deletes/changes a Print Job with App2Me Manager.
Scan Data ・Scanned File	Widget	Get Delete	Requests App2Me Manager to retrieve the scan results. Unnecessary results are deleted.

(2) Between App2Me Manager and App2Me Provider (Modified MDNS, HTTP)

App2Me Manager and App2Me Provider communicate with each other in order to transfer print or scanned data, and will also actively search for partners to communicate with.

The following data is exchanged:

※All data is exchanged using HTTP unless otherwise noted.

Data	Originator/Sender	Control/Request	Flow
User Information ・Manager URL ・Manager version information ・User Name ・Client PC Host name	App2Me Manager	Notification (Modified MDNS)	User information is sent to all devices that are running App2Me Provider.
Search Query	App2Me Provider	Search (Modified MDNS)	App2Me Provider requests all App2Me Managers to resend the user information.

Device Information ・Device Name ・Device IP	App2Me Provider	Notification (Modified MDNS)	App2Me Provider sends the indicated device information to App2Me Manager.
Widget Information ・Widget Name ・Widget Icon ・Widget PIN data ・App2Me Manager Password	App2Me Provider	Get	This information is retrieved by App2Me Provider from App2Me Manager. The PIN and Password data are hash values are used for HTTP digest authentication.
Printing Job ・ Job data ・ Print settings	App2Me Provider	Get Delete	App2Me Provider retrieves the print job from App2Me Manager, and then deletes the completed job.
Scan Data ・Scanned File	App2Me Provider	Add	The Scan results are sent to App2Me Manager.

(3) Between App2Me Manager and the Update Server.
In order to update App2Me Manager, a connection to the Update Server must be made. The following data is exchanged:

Data	Originator/Sender	Control/Request	Flow
App2Me Manager Object	App2Me Manager	Get	App2Me Manager will download the latest version of App2Me Manager, if an update module is available.

(4) Between App2Me Manager and the Log Server
App2Me Manager is set to transfer the Usage Log to the Log Server by default. This option can be changed (turned off/on) using the App2Me Manager Configuration Settings.

Data	Originator/Sender	Control/Request	Flow
・Usage Log ・Log Metadata	App2Me Manager	Add	App2Me Manager sends the Usage Log to the Log Server at pre-determined intervals.

■Log Metadata

Information	Content
Country Information	Country information where App2Me Manager is running.
GUID	The App2Me Manager GUID that is created during installation.

OS Information	OS version information for the client PC running App2Me Manager is included in the report.

■Usage Log

Information	Content
Date and Time	Day and Time the log was output. Time is indicated using GMT.
Event	The following App2Me Manager operations, such as Widget registration, jobs registration, etc. are listed as events: ・Widget Registration ・Job Registration ・Scan Registration ・Print Job Data Information ・Scanned Data Information ・Direct Printing Request.
Widget ID	Each Widget contains a unique ID.
Widget Type	Widgets have two types: ・Scan ・Print
Result	Records the success or failure of an event.
Device ID Number or Model Code	If possible, the device ID number or model code is collected from the device that was used with App2Me Provider.

4.2. Data Storage

This section explains what data is stored, where is stored and how App2Me handles the information.

(1) App2Me Manager

There are 2 directories used to store App2Me data on the client PC:

a) User directory (Home Folder)

The Data stored in this area is only accessed by the user and its content is protected by file system permissions that are generally regarded as safe.

The App2Me Manager Application preferences (Configuration Data) and Usage log are stored in this directory.

b) Temporary Folder

The location of this directory is set by the operating system environment variable TEMP. The temporary folder is **%USERPROFILE%¥Local Settings¥Temp** by default, but the user is able to modify this setting. This directory is used to hold the temporary files App2Me creates in order to carryout its tasks, and the data is deleted every time App2Me is started.

(2) App2Me Provider
App2Me Provider stores port settings, the hash value of the Administrator PIN, and the automatic logout time setting locally. This data cannot be accessed without the Administrator PIN, so it is thought to be safe.

5. Authentication

This section explains the Authentication functions used by the App2Me components.

5.1. App2Me Manager
It is possible to set a password (128-byte maximum) into App2Me Manager in order to restrict access to a user's App2Me Manager. This password is input when that user is selected on a device.

5.2. Widget
A 10-character password can be stored with each Widget. This password is required when that widget is selected on the device. This password is independent from the App2Me Manager Password, and a different password can be set to for each widget.

5.3. App2Me Provider
App2Me Provider has a special an administrator settings menu that is accessed by a special password. This default password is "0000", but up to 8 digits can be used. This administrator settings password is independent from the device administrator account password and is only for App2Me Provider. The password is changed by logging into App2Me Provider and using the administrator settings.

6. Security Considerations

This section explains how data is protected by App2Me.

6.1. Document Data
Data exchanges between App2Me Manager and App2Me Provider (print jobs and scan data transfer), do not use HTTPS or another secure connection method. It is therefore recommended that important documents by sent using App2Me when the internal network environment is considered secure.

6.2. App2Me Provider Data Settings
As App2Me Provider Data Settings are stored on the device's HDD, it is possible to protect this data using the device HDD Encoding Option.

6.3. App2Me Manager Data Storage
App2Me Manager's settings are stored in the user directory, and are protected using the OS's built-in user access control system. But as print job data is stored in the shared temporary files folder it is not recommended that App2Me be used to send important documents to a shared PC.

6.4. Usage Log

The Usage Log contains usage information about App2Me gathered from the user's activities, but user names and other identifying information are not included. In addition, the Usage Log can be completely deactivated if desired.

6.5. App2Me Manager Update

App2Me Manager does not possess a mechanism for verifying the validity of the data downloaded for update purposes. Due to this, it is recommended that App2Me not be run on public networks.

7. Notes about Using App2Me

7.1. About the changes to the App2Me modified MDNS port

While the modified MDNS's port number is variable, the standard MDNS port 5353 cannot be used. If a PC or device is using both the standard and Ricoh modified versions of MDNS, there can be a port conflict and communication errors.

8. FAQs

8.1. What changes were made to Ricoh's modified MDNS?

MDNS was selected as the communications method for giving App2Me Provider the ability to search for App2Me Manager (and vice versa).

The primary reasons for this were:

- MDNS is a multicast protocol for auto discovery that is being considered for RFC standardization, and that Apple has already adopted as standard technology.
- In comparison with other type of multicast protocols for auto discovery (WSD, UPnP, etc), MDNS is not OS-dependent, and the amount of data is light. (WSD and UPnP use XML data; MDNS uses text data)
- Since the tasks that require App2Me Manager and App2Me Provider communicate are usually small (job data exchange, etc.), it is felt that an IP address exchange is not necessary.

The reasons that MDNS was modified are:

- (1st Modification) Unicast communication was added.
  Unicast was specifically added to allow the finding and searching of App2Me Providers and App2Me Managers on different subnets.

- (2nd Modification) Announcement Function Control
  A control was placed on the amount of announcements from both App2Me Provider and App2Me Manager in order to reduce the amount of unnecessary traffic on the network.
  Only the following information is announced:
  ○ Own IP address

- ○ Own Port Number
- ○ User Name used in App2Me Provider
- ○ The URL of the Widgets in the App2Me Manager

- (3rd Modification) Addition of Port Number Selection function
  Port selection was added to allow users to select the port number in order to prevent network conflicts if the user is running the normal MDNS protocol.

- (4th Modification) Control to prevent DNS records to be delivered.
  Reason: Due to the record A/AAAA can substitute another record, then it is deleted.(In normal circumstances this record is compatible with the IP address and the Host name. Due to App2Me data exchange is IP based this is unnecessary)

- (5th Modification) Added auto discovery services types for App2Me Manager and App2Me Provider and modified the DNS record.
  Reason: There was not support for App2Me services type in the RFC standards.
  (RFC link http://www.dns-sd.org/ServiceTypes.html)

8.2. How much network traffic does the modified MDNS generate?

There are 4 situations in which the modified MDNS generates network data. These are merely per-client, per-device examples. The actual amount will depend on how many devices and clients are running App2Me.

A) When either App2Me Provider or App2Me Manager attempt discovery, one DNS PTR record (100 bytes approximately) is sent 3 times.

B) App2Me Provider/App2Me Manager will respond to A) with the DNS PTR record, a SRV record, and a TXT record for each PTR record received. The total data is approximately 200 bytes per response.

C) When the [List of Available Devices] is accessed on App2Me Manager, App2Me Manager attempts auto discovery. App2Me Manager sends out a SRV record (100 bytes, approximately) 3 times. This SRV record requests the model name and network address of the responding device.

D) App2Me Provider replies to the App2Me Manager from C) with a response that includes the SRV record and a TXT record for each SRV record received(approximately 200 bytes per response).