

[APNOTE05]

携帯電話から POP サーバーメールを転送受信

ABS-9000 DeviceServer

APNOTE05 Rev A.1.1

2008/10/18



オールブルーシステム (All Blue System)

ウェブページ: www.allbluesystem.com

コンタクト: contact@allbluesystem.com

1 イン트로ダクション

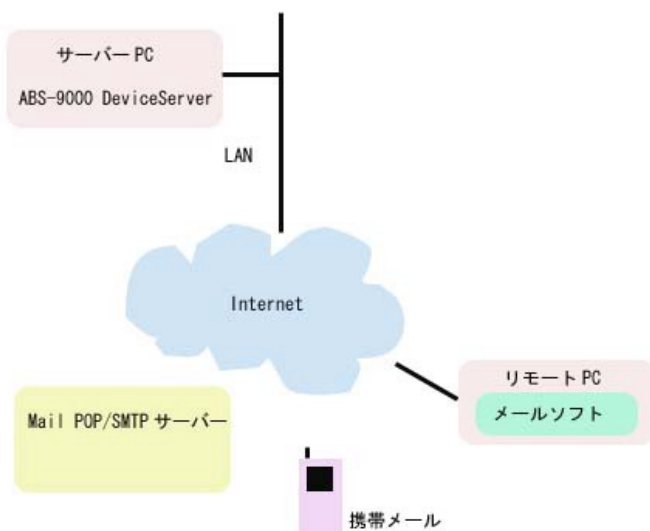
携帯電話メールやリモートPC（インターネットメールにアクセス可能なPC）のメールクライアントソフトから、DeviceServer にメール転送リクエストを送信して、DeviceServer で管理しているPOPメールサーバーに保管されたメール内容を読み込んで、転送するシステム例について説明します。

リクエストを送信した時に、POPサーバーのメールボックスにある全てのメールのヘッダ情報とメール本文（最初から決められた行数まで読み込んだもの）を1つにまとめて、リクエストを発行したメールクライアント（携帯電話メールやリモートPC） に送信します。

2 必要な機材・リソース

必要なシステムやデバイス等	説明
ABS-9000 DeviceServerの動作しているPC	スタンダードライセンスもしくはエンハンスライセンスが必要になります。
POP メールアカウント	メールを受信するPOPサーバーメールアカウント
SMTP メールアカウント	メールを送信するSMTPサーバーメールアカウント

3 システム構成図



4 システム動作概要

- リモートPCのメールソフトや、携帯電話メールから DeviceServer で管理しているPOP サーバーのメールアカ

ウントに、メールコマンド（\$LOGIN\$）を送信してログイン認証を行う。その後、セッショントークンのリプライメールを受け取る。

- リモートPCのメールソフトや、携帯電話メールからスクリプト実行のメールコマンド（\$SCRIPT\$）を送信する。そのときに、ログイン認証で得られたセッショントークンとメール転送スクリプトの名前（MAIL_DUMP）、1メールあたりの本文を読み込む最大行数をメール本文に指定する。
- DeviceServer で MAIL_DUMP スクリプトが実行される。POPサーバーにある全てのメールのメールヘッダとメール本文を一つにまとめて、リモートPC や携帯にメールで送信する。スクリプト実行自身の実行結果ステータスのメールも別に送信する。

5 設定手順

5.1 サーバー設定

サーバー設定プログラムで、下記の項目を設定します。

サーバー設定プログラム	
設定が必要な項目	設定内容
メール機能を有効にする	チェックを付ける
受信メールをチェックする	チェックを付ける
メール確認間隔	3分。（プロバイダのPOPサーバーの設定によって、メール確認間隔を調整してください）
新しいメール受信時にスクリプト実行	チェックを付ける
POPサーバー	プロバイダのPOPサーバーの設定に合わせて下さい
POPユーザー	プロバイダのPOPサーバーの設定に合わせて下さい
POPパスワード	プロバイダのPOPサーバーの設定に合わせて下さい
POPポート	プロバイダのPOPサーバーの設定に合わせて下さい
自分のメールアドレス	DeviceServer からメール送信するときのデフォルトメール送信元アドレスを記入します。
SMTPサーバー	プロバイダのSMTPサーバーの設定に合わせて下さい
SMTPユーザー	プロバイダのSMTPサーバーの設定に合わせて下さい
SMTPポート	プロバイダのSMTPサーバーの設定に合わせて下さい

5.2 スクリプト設定

5.2.1 MAIL_DUMP スクリプト作成

POPサーバーにある全てのメールのメールヘッダとメール本文を一つにまとめて、リモートPC や携帯電話にメールで送信するスクリプトを作成します。

このスクリプトは、メールコマンドで実行されることを想定して、リプライメール先のアドレスはスクリプトパラメ

ータ g_params["\$REPLYTO\$"] から取得しています。スクリプト実行時に指定されたパラメータ "Lines" で 1 メールあたりに読み込む最大行数を指定できます。パラメータ "Lines" を指定しなかった場合は 30 がデフォルトで指定されます。

POP サーバーのメールは、このスクリプト実行した後もそのまま残りますので、後からメールクライアントソフトで再び読み込むことができます。

ファイル名 (MAIL_DUMP.lua) で DeviceServer のスクリプトフォルダに保管します。

```
file_id = "MAIL_DUMP"

-----
-- BEGIN SCRIPT --
-----

log_msg("start..", file_id)

local body = {}
local max_lines

if g_params["Lines"] then
    max_lines = tonumber(g_params["Lines"])
else
    max_lines = 30
end

-----
-- 現在の POP サーバーのメールの内容を取り込んで、1つにまとめてメールで送信する。
-- スクリプトパラメータ (Lines) が指定されていた場合は、各メール毎に最初から
-- 最大 Lines で指定された行までの内容だけを本文として取り込む。指定されていない場合は
-- デフォルトで、30 が指定される。
-- このスクリプトはメールコマンドから実行されることを想定しています。
-----

local stat, timestamp, from, recipients, subject, priority, id = mail_retrieve_header()
if not stat then error() end
for key, val in ipairs(id) do

    -- メールコマンドは除外する
    if (not string.match(subject[key], "^%$%u+%$$")) then
```

```

table.insert(body, "Date:" .. timestamp[key])
table.insert(body, "Subject:" .. subject[key])
table.insert(body, "From:" .. from[key])
table.insert(body, "Body:")

local each_stat, each_header, each_body = mail_receive(val, false)
if not each_stat then error() end
if each_header["MissedParts"] == "No" then
    if (#each_body > max_lines) then
        for cntr = 1, max_lines, 1 do
            table.insert(body, each_body[cntr])
        end
        table.insert(body, "")
        table.insert(body, "continue...")
        table.insert(body, "")
    else
        for ekey, eval in ipairs(each_body) do
            table.insert(body, each_body[ekey])
        end
        table.insert(body, "")
        table.insert(body, "-- end of mail --")
        table.insert(body, "")
    end
end
end
end
end

-----
-- メール送信 --
-----

stat = mail_send(g_params["$REPLYTO$"], "", "メールダンプ", unpack(body))
if not stat then error() end

-----
-- END SCRIPT --
-----

```

 **注意**

スクリプト中に日本語を記述するときは、スクリプトファイルを UTF-8N 形式で保存してください。Shift_JISや UTF-8 BOM付き形式などで保存すると、DeviceServer でエラーが発生します。Windows付属のワードパッドやメモ帳ではこの形式で保存できませんので、別途 UTF-8N 形式で保存可能なエディタソフト (*1) を使用してください。

(*1) TeraPad 等のソフトウェアがよく使用されています。

6 メール転送実行例

ここでは、実際に転送を行うメールを送信した時の例を挙げます。メールコマンドの機能については“DeviceServer ユーザーマニュアル”を参照して下さい。

- Step1 ログインメール送信

メール送信	
メール宛先	your_mail_address@your_mail_domain
メール件名	\$LOGIN\$
メール本文	user_name user_password

メール宛先は、DeviceServer で設定したPOPサーバーのメールアドレスを指定します。

user_name, user_password には、DeviceServerのユーザー情報に登録されたものを使用します。ユーザー設定で、メールコマンド応答先が設定されている場合には、送信を行った携帯やリモートPC のメールアドレスがメールコマンド応答先と一致していない場合は、DeviceServer からのリプライメールを受け取る事ができませんので注意してください。

- Step2 ログイン認証成功、セッショントークンメール受信

メール受信	
メール件名	\$REPLY\$ -LOGIN SUCCESS-
メール本文	ログイン成功 ST02296B192B391F

2行目の文字列がセッショントークンです。

ログインに失敗した場合はエラー内容が記載されてメールが送られてきます。

- Step3 転送スクリプト実行メール送信

メール送信	
メール宛先	your_mail_address@your_mail_domain
メール件名	\$SCRIPT\$
メール本文	ST02296B192B391F MAIL_DUMP Lines 10

メール宛先は、DeviceServer で設定したPOPサーバーのメールアドレスを指定します。

メール本文の1行目にはStep2 で取得したセッショントークンを記入します。

3行目以降はスクリプトパラメータを指定します。ここでは、“Lines” を指定して、1メールあたりの読み込み最大行数を指定します。その後ろにスペースで区切って数字を指定します。“Lines” スクリプトパラメータが省略されたときは、30 がデフォルトになります。

- Step4 スクリプト実行成功メール受信

メール受信	
メール件名	\$REPLY\$ -SCRIPT SUCCESS-
メール本文	スクリプト実行成功

スクリプト実行に失敗した場合はエラー内容が記載されてメールが送られてきます。

- Step5 転送メール受信

メール受信	
メール件名	メールダンプ
メール本文	Date:2008/10/1 16:22:54 Subject:xx From:xx Body: xx xx xx xx xx xx xx xx xx xx xx xx xx

```
continue...

Date:2008/10/1 16:22:54
Subject:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
From:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Body:
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

-- end of mail --

Date:2008/10/1 16:22:54
Subject:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
From:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Body:
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

continue...
```

7 このドキュメントについて

7.1 著作権および登録商標

Copyright© 2008 オールブルーシステム
このドキュメントの権利はすべてオールブルーシステムにあります。無断でこのドキュメントの一部を複製、もしくは再利用することを禁じます。

7.2 連絡先

オールブルーシステム (All Blue System)

ウェブページ <http://www.allbluesystem.com>

メール contact@allbluesystem.com

7.3 このドキュメントの使用について

このドキュメントは、ABS-9000 DeviceServer の一般的な使用方法と応用例について解説してあります。お客様の個別の問題について、このドキュメントに記載された内容を実際のシステムに利用するときには、ここに記載されている以外にも考慮する事柄がありますので、ご注意ください。特に安全性やセキュリティ、長期間にわたる運用を想定してシステムを構築する必要があります。

オールブルーシステムでは ABS-9000 DeviceServer の使用や、このドキュメントに記載された内容を使用することによって、お客様及び第三者に損害を与えないことを保証しません。ABS-9000 DeviceServer を使用したシステムを構築するときは、お客様の責任の下で、システムの構築と運用が行われるものとします。