OTA (Over The Air) Overview

Hanyang University
CASP LAB
02-2296-0390
Http://casp.hanyang.ac.kr
Contents

- OTA-Http Overview
- IP Connectivity Procedure
- TCP Connection Procedure
- Terminal Registration
- Identification (RFC 2617, Client Authentication)
  - Unauthenticated Identification
  - Authenticated Identification
- Content Push
- Session Initiation Request
- Service Indication, Service Loading
OTA-HTTP Overview

- Http 1.1 with TCP/IP bearers (e.g. GPRS)
- Support only connection-oriented push
  - Terminal (HTTP server), PPG (HTTP client)
- Core features
  - IP connectivity procedure
    - This protocol is designed to work with Network(Terminal) initiated IP connectivity establishment, rely upon terminal’s ability to establish
  - TCP connection procedure
    - TO-TCP connection establishment
    - PO-TCP connection establishment
  - Terminal registration procedure
    - Always initiated by the PPG, which becomes aware of the terminal’s current capabilities and preferences
  - Content push
    - Use HTTP’s POST method
  - Identification and optional authentication
OTA-HTTP IP Connectivity Procedure

- Terminal-initiated IP connectivity establishment is trivial. (like PPP Connection)
- Network-initiated IP connectivity establishment
- Find out the terminal’s IP address
  - Use static IP addresses
  - Lookup in a RADIUS server
  - PPG sends an SIR to the terminal using either connectionless push over a bearer where a well-known address can be used (MSISDN for SMS), or by using connection-oriented push if applicable.
Active TCP connections

- Connections used for registration and push delivery
- TO-TCP connection establishment method
  - Both terminal and PPG must support the non-secure TO-TCP
- PO-TCP connection establishment method
  - Terminal must support the non-secure PO-TCP and PPG should support the non-secure PO-TCP
- Either the PPG or the terminal may at any time close an active TCP connection.
• Imply that terminal prepared to receive HTTP Request

DynP = Dynamically assigned port
ProvP = Provisioned port
RegP = Registered port (sec/nonsec)
SirP = Port in SIR

The TO-TCP method
OTA-HTTP TCP Connection Procedure (3/3)

Assume the terminal has IP connectivity and its IP address is known to PPG

The PO–TCP method
PPG should carry out the registration procedure
- when an active TCP connection has been established in order to identify/authenticate the terminal and find out about its capabilities and preferences.

Registration context
- Defined within the scope of a certain Terminal-ID, and also the bearer used when the CPI was conveyed.
- PPG stores multiple identifiable CPIs for a registration context, where each CPI is identified by a CPITag computed by the terminal.
OTA-HTTP Terminal Registration (2/5)

Registration request (HTTP OPTIONS)

Terminal

PPG

OPTIONS /wappush HTTP/1.1
Host:

HTTP OPTIONS, X-Wap-CPITag=X

204 No Content, X-Wap-CPITag=Y, CPI Headers
X-Wap-Push-Status=501, Accepted, CPITag mismatch

HTTP OPTIONS, X-Wap-CPITag=Y

204 No Content
X-Wap-Push-Status=500, Accepted, CPITag match

CPITag=X

CPITag=Y

CPITag=Y
OTA-HTTP Terminal Registration (3/5)

**Registration Validation (HTTP POST for push request))**

- **Terminal**
  - POST /wappush HTTP/1.1
  - Host:
  - HTTP POST, X-Wap-CPITag=X, push content
  - 204 No Content, X-Wap-CPITag=Y
  - X-Wap-Push-Status=256 Rejected, CPITag mismatch

- **CPI with CPITag=Y Available?**
  - Yes
  - No
  - Registration Request

- **PPG**
  - CPI with CPITag=Y Available?
  - Yes
  - No

- **PPG**
  - CPI with CPITag=Y Available?
  - Yes
  - No
  - Registration Request
OTA-HTTP Terminal Registration (4/5)

- X-Wap-CPITag
  - Carry a CPITag value between PPG and terminal, which represents a specific set of CPI header values
  - Terminal must re-compute the CPITag value each time one or more CPI headers change

- CPITag value
  - Concatenate all CPI Header values in the response
  - Apply a hash algorithm that generate at least a four octet hash on the concatenated value. (SHA-1)
  - Use the first four octets of the output
  - Must use Consistent concatenation manner
CPI headers

- X-Wap-Push-Accept:
  - Default: application/vnd.wap.sia, text/vnd.wap.si
- X-Wap-Push-Accept-Charset: UTF-8
- X-Wap-Push-Accept-Encoding: identity
- X-Wap-Push-Accept-Language: *
- X-Wap-Push-Accept-AppID: *
- X-Wap-Push-MsgSize: 1400
- X-Wap-Push-Accept-MaxPushReq: 1
OTA-HTTP
Terminal Registration-SHA Algorithm

512 bit

Concated Data | Padding | Length of data

A = 0x67452301
B = 0xefcdab89
C = 0x98dadcfe
D = 0x10325476
E = 0xc3d2e1f0

A

B

C

D

E

f1

CLS5

CLS30

CPITag

Value
After Connection establish, PPG and Terminal must identify mutually

Authentication may be used security channel according RFC2617

Authentication Scheme is not restricted in first HTTP request

PPG use Terminal ID, convey Terminal ID follow rule
  - If terminal support WAP Prov, use PXAUTH-ID
  - If not support or fail to select PXLOGICAL, Terminal-ID must be formatted according Client ID

Unauthenticated identification

Authenticated identification
OTA-HTTP Identification & Authentication

- HTTP Authentication (RFC 2617)
  - HTTP Server authenticate HTTP Client to access specified request URI
  - Recommended Security Channel
  - Common Authentication scheme
    - Basic, Digest
  - Server challenge
    - WWW-Authenticate: Basic realm=“KIMHYS’sWORLD”
  - Client credential
    - Authorization: Basic base64(userid:passwd)
- HTTP Server authenticate client id and password
Unauthenticated Identification
- Terminal must include Terminal ID in response to OPTIONS request (Registration request)
- Terminal not authenticate PPG and send Terminal ID
  - So called “Unauthenticated Identification”
- Using X-Wap-Terminal-Id header
- PPG receiving Terminal-ID, authenticate terminal
- If can not use Terminal-ID
  - PPG request to terminal using X-Wap-Authenticate header
  - Terminal ID conveyed through X-Wap-Authorization header
  - Terminal can request PPG to authenticate itself (Authenticated)
OTA-HTTP Unauthenticated Identification

HTTP OPTIONS, X-Wap-Authenticate

204 No Content, X-Wap-Authorization, OPTIONS response
X-Wap-Push-Status=500/501 Accepted

X-Wap-Authenticate: Digest realm="ppg identity",
qop="auth, auth-int"
nonce="dcd930560dkdfalkdf345…"
opaque="asdfwer445346asdf…"

HTTP POST, push content

204 No Content
X-Wap-Push-Status=400/401 Accepted

Terminal does not authenticate PPG

Terminal authenticated

PPG

Terminal

Hanyang Univ. CASP lab.
OTA-HTTP Identification & Authentication

- Authenticated Identification
  - PPG and Terminal support ‘basic’ and ‘digest’ authentication scheme according to RFC2617

- Authentication
  - PPG to authenticate Terminal, send X-Wap-Authenticate header and analyses terminal X-Wap-Authorization header
  - Terminal to authenticate PPG, response with status code 401, WWW-Authenticate header and analyses Authorization header from PPG
OTA-HTTP Authenticated Identification

HTTP OPTIONS, X-Wap-Authenticate

401 Unauthorized, X-Wap-Authorization, WWW-Authenticate

X-Wap-Push-Status=300 Rejected

HTTP OPTIONS, Authorization

204 No Content, OPTIONS response

X-Wap-Push-Status=500/501 Accepted

Terminal

PPG

WWW-Authenticate: Digest
username="TerminalID",realm="ppg identity",nonce="dcd930560dkdfalkdf345...",domain="/wappush",
qop=auth,opaque="asdfwer445346asdf..."

HTTP OPTIONS, X-Wap-Authorization

204 No Content, OPTIONS response

X-Wap-Push-Status=400/401 Accepted

Authorization: Digest username="ppg identity",realm="ppg identity",
nonce="dcd930560dkdfalkdf345...",digest-uri="/sec/sec2",qop=auth,opaque="asdfwer445346asdf..."

HTTP POST, push content

204 No Content

X-Wap-Push-Status=400/401 Accepted

Terminal authenticated

PPG authenticated
OTA-HTTP example of Basic Authentication

HTTP OPTIONS, Authorization, X-Wap-Authenticate

204 No Content, X-Wap-Authorization, OPTIONS response
X-Wap-Push-Status=500/501 Accepted

Authorization: Basic "qM345A56abpzqe5=="
Data of base64 encoded identity of ppg

Terminal authenticated

HTTP POST, push content

Authorization: Basic "qM344tyq5zqe7=="
Data of base64 encoded terminal ID

PPG authenticated

204 No Content
X-Wap-Push-Status=400/401 Accepted
OTA-HTTP Application Addressing

- **PPG** must address the terminal push application using `/wappush abs_path` as the URI of the POST request.

- **Terminal** must use `X-Wap-Application-Id` value to route the push request to the intended application.

- If no `X-Wap-Application-Id` header, terminal must assume the **WML User agent**.
OTA-HTTP Content Push(1/3)

- HTTP POST Request Format
  - Message body carries content and headers
  - Status-line in entity body contains code legal for an HTTP response
  - Can be included, X-Wap-Push-ProvURL, X-Wap-Push-Info header in POST request
OTA-HTTP X-Wap-Push-Info

- A request header in POST request to indicate follow push transaction
  - authenticated : used as authenticated flag
  - trusted : used as Trusted Flag
  - last : used as last flag
  - response : indicate message body to be included in POST response

X-Wap-Push-Info: authenticated, trusted
X-Wap-Push-Info: trusted, last
X-Wap-Push-Info: response
Post Request Format

POST /wappush HTTP/1.1
Host:
Content-Type: application/http
X-Wap-Push-ProvURL: http://www.terminal.url
X-Wap-Push-Info: authenticated

HTTP 200 OK
...
Message content
POST Response Format

- Response to POST request contain status line outcome of request
- Status code
  - 204 “No Content”: response don’t contain message body
  - 200 “OK”: response contain message body
- X-Wap-Push-Status
  - Reflection outcome of registration request or push request
OTA-HTTP Push Message Example

POST /wappush HTTP/1.1
Host:
Date: Tue, 31 Jul 2001 10:13:05 GMT
Content-Type: application/http
Content-Length: 504
X-Wap-Push-OTA-Version: 1.0

HTTP/1.1 200 OK
Content-Language: en
Content-Type: text/vnd.wap.si
X-Wap-Application-Id: x-wap-application:wml.ua

HTTP/1.1 204 No Content
X-Wap-Push-Status: 400 Accepted
Push is async, it is possible that no push session or active TCP connection exist

PPG request SIR to SIA on Terminal to establish WSP session or TCP connection

SIR use connectionless or connection oriented push

SIA Content contains two type contact points for OTA-HTTP, OTA-WSP

PPG and SIR can use one of them and both, and Terminal MUST use only one of them
OTA-Session Initiation Request (2/2)

- **SIR in OTA-HTTP**
  - SIA must be supported terminal and PPG
  - Terminal procedure
    - Establish IP connectivity, if not
    - Proceed with TO_TCP connection
  - Multiple contact points, establish each contact points

- **SIR in OTA-WSP**
  - Terminal procedure
    - Establish connectivity with the network
    - Establish sessions towards contact points
OTA-SIA Content Based PDU (1/3)

- Content-type: application/vnd.wap.sia
- Version: current 1, must be compatible 0
- *Len: length following field
OTA-SIA Content Based PDU (2/3)

- Application-ID list
  - APP ID on terminal when push session established, registration

- Contact Points WSP
  - List of server address for push session

- Contact Points
  - List of server address for other protocol than WSP

- ProtOpts : contain a list to identify protocol
  - First identifier identify protocol used when contacting first contact point, and so on
OTA-SIA Content Based PDU (3/3)

- **ProtOpts**

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OTA-HTTP, no CPITag present</td>
</tr>
<tr>
<td>1</td>
<td>OTA-HTTP, CPITag present</td>
</tr>
</tbody>
</table>

- **CPIITag**
  - First CPIITag map to first contact point if CPIITag is present, and so on
  - Not match ProtOpt number and CPIITag number, first CPIITag apply all contact points
OTA-HTTP Connection Oriented Push

Client device

Push Proxy Gateway

SIR

Generic

Wap

SI A Content

Pom-SessionRequest.ind

SI A

CPI Tag computing
With SHA-1

Pom-SessionRequest.req

TO-TCP establishment

Http OPTIONS

Registration request

Http

CPI Tag=x
CPI Headers

X-Wap-Application-Id:
x-wap-application:push.sia

Content-Type:
application/vnd.wap.sia

X-Wap-CPITag: xxxx

X-Wap-Push-Info:
Authenticated, trusted

X-Wap-Application-Id:
x-wap-application:wml.ua

may include
Authentication scheme

Registration establish
Service Indication (1/2)

- Application on OTA in Terminal
- Provide to send notification to end-user in asynchronous manner
- Provide the choice to start the service immediately or to postpone
- Used new mail, stock price, and so on
- Textual form:
  - text/vnd.wap.si
- Tokenised form:
  - application/vnd.wap.sic
Service Indication (2/2)

1. Push Initiator
2. Push Proxy/Gateway
3. Mobile Client
4. Method Proxy/Gateway
5. Origin Server

- You have 4 new E-mails
- Retrieve, Postpone
- E-mail service.
  - >Read
  - >Send
  - >Delete
  - >Options

- HTTP GET
- WSP GET
- "HTTP GET"
- "WSP GET"
- WML (textual)
- WML (binary)
Service Loading(1/2)

- Application on OTA in Terminal
- Provide ability to UA to load and execute a service without user intervention
- Provide followed basic functionality
  - Control of the level of user-intrusiveness
    - If client is busy, control loading the service
  - Pre-emptive content caching
    - Indicated content is downloaded and cached by setting attribute of SL
Service Loading(2/2)

Push Initiator

1

Push Proxy/Gateway

2

SL (textual)

SL (binary)

Origin Server

“HTTP GET”

WML (textual)

Method Proxy/Gateway

“WSP GET”

WML (binary)

Mobile Client

“Idle”

Your prepaid Balance is low.

Reload

Ignore

Call Customer Care

Hanyang Univ. CASP lab.
OTA (Client Side)

Presentation

SIA

SI

SL

SIA Content

HTTP

WSP

TCP

WTP

HTTP_Header

“Accept Application”

“Bearer Indication”

…

S-Connect

Pom-Connect

TO-TCP request